RICE UNIVERSITY

Deformations of Simply Periodic Scherk-Type Minimal Surfaces

by

Matt McLelland

A Thesis Submitted
in Partial Fulfillment of the
Requirements for the Degree

Doctor of Philosophy

Approved, Thesis Committee:

Michael Wolf, Chairman
Professor of Mathematics

Robert Hardt
W.L.Moody Professor of Mathematics

Mark Embree
Associate Professor of Computational and
Applied Mathematics

Houston, Texas

April, 2009

Deformations of Simply Periodic Scherk-Type Minimal Surfaces

Matt McLelland

Abstract

We demonstrate that a certain dimensional family of symmetric singly-periodic minimal surface with Scherk-type ends exists in the neighborhood of a given example if a set of holomorphic quadratic differentials is independent. Our approach extends the work of Traizet, who has previously shown the existence of a family of such minimal surfaces in the neighborhood of a degenerate example consisting of a number of intersecting planes. Whereas in Traizet's construction the underlying conformal structure was a union of Riemann spheres, we treat the case where the underlying conformal structure is a Riemann surface of higher genus. In our approach, admissible surfaces are identified with Weierstrass data satisfying certain constraints. Using the bilinear relations and the Rauch variational formula, we are able to find holomorphic quadratic differentials which represent differentials of the constraints, and whose independence, by an implicit function theorem argument, implies the existence of the desired surface family in a neighborhood of the original. We restrict our attention to tori and develop machinery for investigating the quadratic differentials numerically using interval arithmetic to obtain provable bounds on their residue structure. The developed tools are finally applied to an example surface in Karcher's one-dimensional toroidal saddle tower family, which is shown to exist in a larger three-dimensional family.

Acknowledgements

Foremost, I thank my thesis advisor Mike Wolf for his invaluable guidance and vast patience. I also thank the other members of my thesis committee, Bob Hardt and Mark Embree for their helpful feedback and patience. I thank Maxine Turner and Marie Magee for their enormous assistance and for their patience. Finally, I thank my wife, my parents, and my wife's parents for their continual support and patience.

Contents

List of Figures

Chapter 1

Introduction

It was known to Euler that the only surface of revolution critical with respect to the area functional is the catenoid, obtained by revolving the graph of $f(x) = cosh(x)$ around the x-axis. However, here 'critical' means only with respect to variation of f or only among rotationally symmetric surfaces. One can also ask whether such a surface is additionally critical for area with respect to more general (potentially non-symmetric) variations, leading to the definition of a minimal surface. In the 1770s, Meusnier gave a geometric description of minimal surfaces as surfaces of zero mean curvature $H = \frac{1}{2}(k_1 + k_2)$, where k_1 and k_2 are principal curvatures, and was able to prove that both the catenoid and helicoid were minimal. The plane, the catenoid, and the helicoid (or sub-surfaces thereof) remained the only known examples of minimal surfaces until 1835, when Scherk discovered a doubly periodic surface now known as Scherk's first surface. Not long after, he discovered Scherk's second surface which shall play a prominent role in this work.

Later in the nineteenth century, connections between minimal surfaces and harmonic functions began to be noticed, which paved the way for studying minimal surfaces using the tools of complex function theory. The period from 1850 to 1880 has been called the golden age of minimal surfaces. During this period, much progress was made, but even through to the early twentieth century, the focus of the great majority of mathematicians studying minimal surfaces was toward a solution of the Plateau problem, which asks to find the minimum area surfaces spanning a given

boundary. In the last thirty years, following the discovery by Costa of a new minimal surface with finite topology, there has been a wealth of new results pertaining to minimal surfaces and in particular to complete and embedded minimal surfaces of finite genus.

In [Karcher88], Karcher constructs a number of minimal surfaces based on or generalizing Scherk's examples. In particular, he constructs a family of toroidal saddle towers that will be used as the example on which the techniques of this work are tested, and will be the subject of the main theorem of this work (see Figure 1.1). The family is parameterized by the order of its rotational symmetry group ($N \geq 3$) and the modulus of the underlying torus (τ). His construction of this surface family uses elliptic function theory to show that the conditions required for a minimal surface are satisfied automatically in the presence of sufficient symmetries, except for a single real condition which he is able to prove solvable using an intermediate value theorem argument. From geometrical considerations, Karcher is able to identify fundamental domains of the symmetries of the surface, and deduce from their boundaries being graphs over a plane that the pieces, and thus the whole surfaces, are embedded.

In [Kapouleas96], Kapouleas uses tools from partial differential equations to desingularize the intersections of two minimal surfaces, creating a new minimal surface that hueristically looks similar to the union of the original surfaces away from the intersection and at the intersection consists of a finite (but large and unknown) number of tiny handles.

In [Traizet96], Traizet applies the techniques of Kapouleas to glue a number Scherk surfaces together at their ends. The result can be described as the desingularization of a set of vertical planes where the intersections look like Scherk. In the follow-up work [Traizet00], Traizet gives another construction of this surface family, this time using techniques of algebraic geometry to "open the nodes" of a Riemann surface with

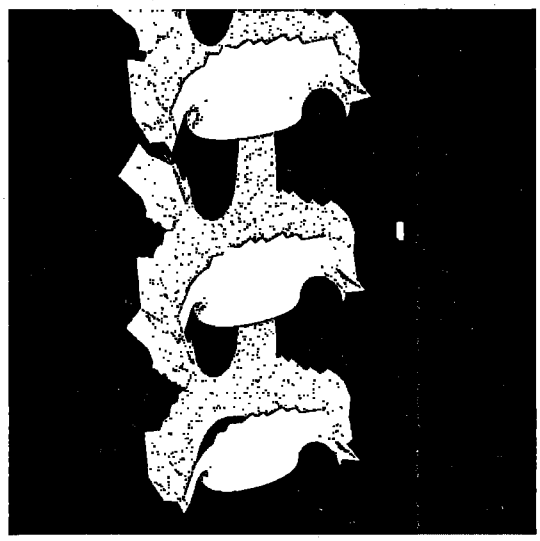

Figure 1.1 : Karcher's surface - a partial rendering of the surface using the software developed for this thesis

ordinary double points (nodes). This new approach allows him to obtain Weierstrass representations. Traizet's family of surfaces overlaps the Karcher family of toroidal saddle tower surfaces. The overlap region consists of Traizet surfaces obtained by desingularizing N planes arranged in a regular N-gon configuration, and consists of Karcher surfaces in a neighborhood of the extreme end of the one-dimensional family where it is asymptoptically approaches a union of planes.

The main theorem of this work, proved in section 4.12, states that,

Main Theorem. *There exists a 3-dimensional family of complete embedded minimal surfaces in a neighborhood of the surface in Karcher's family with 3-fold rotational symmetry and* $\tau = 3i$.

This result suggests that the family constructed in Traizet's work extends all the way to the Karcher surface with $\tau = 3i$ (see Figure 1.1), which was previously known only to exist in the one-dimensional family of Karcher toroidal towers. At a high level,

our proof follows Traizet's approach: identify the minimal desired surfaces with points in a certain parameter domain of Weierstrass data that satisfies some constraints, then show that the differential of the constraint map is full rank and apply the implicit function theorem. But whereas Traizet was able to encode and differentiate constraints using complex function theory (since his base surface is a union of Riemann spheres), we develop our approach on a general Riemann surface and then specialize to a torus to prove our result for Karcher's surface. To work in this general setting we use the tools of Teichmüller theory. We note that [Weber98], [Weber02] and [Hoffman09] have previously employed Teichmüller theory in the construction of minimal surfaces.

In chapter 2 we recall some basic facts from Riemann surfaces, minimal surfaces, and Teichmüller theory. Then, in section 3.1 we develop a basic theory of symmetric and anti-symmetric differentials that will allow us to restrict our parameterization to candidate surfaces that admit a symmetry of reflection. This follows Traizet and effectively halves the dimension of candidate surfaces being considered, but more than compensates for this with the number of constraints that can shown to be automatically satisfied in the presence of a reflective symmetry. In section 3.2, we establish the parameter space we will be considering, and identify the constraints that a point in our parameter space need satisfy if it is to correspond to an admissible minimal surface.

In section 3.3 we come to the first essential ingredient used in finding the constraint differentials, a Rauch variational formula, and in 3.4 we find the other essential ingredient, a bilinear relations formula. In sections 3.5, 3.6, 3.7, and 3.8 we use these tools to find the differentials of our constraints. Finally, in 3.9 we apply a variant of the implicit function theorem to demonstrate that if the obtained differentials are linearly independent at our base surface, then the constraints can all be solved in a neighbhorhood of the base surface, producing a solution set of surfaces in a neighbor-

hood whose dimension is the dimension of the space of differentials less the number of real constraints.

Chapter 4 of this work has no analogue in [Traizet00]. We develop a framework for evaluating functions and differentials on tori to a sufficient degree of precision by finding series representations, explicitly summing a large number of the terms, and then bounding the truncated tails. With the use of interval arithmetic, we are able to compute the residue structure of our set of constraint differentials and convert the question of their independence into a question of the rank of a certain matrix. We construct a computer program to construct the desired matrix and perform Gaussian elimination. The results output by the execution of this program indicate that the matrix is full rank, completing our proof of the main theorem.

Chapter 2

Background

2.1 Riemann Surfaces

A Riemann surface Σ is a surface (two-dimensional manifold) whose transition maps between coordinate charts are all conformal. A function $f : \Sigma \to \mathtt{C}$ is said to be holomorphic (meromorphic) if, for every coordinate chart z, we have that $f \circ z^{-1}$ is holomorphic (meromorphic). A map $f : \Sigma_1 \to \Sigma_2$ between Riemann surfaces is said to be holomorphic if, for every pair of coordinate charts z_1 on Σ_1 and z_2 on Σ_2, the function $z_2 \circ f \circ z_1^{-1}$ is holomorphic on its domain of definition.

We will be primarily interested in the case of Σ compact, possibly with punctures. It will be sometimes be useful to consider such a punctured Riemann surface as a compact surface with distinguished points, and sometimes useful to consider it an open surface. We will usually denote the genus of a Riemann surface as n_g and the homology basis $A_1, B_1, ..., A_{n_g}, B_{n_g}$.

Theorem 2.1.1. (Uniformization) *A compact Riemann surface of genus n_g is conformally diffeomorphic to one of:*

- *a quotient H/Γ where H is th upper half plane and Γ is a discrete subgroup of $PSL(2,\mathtt{R})$, in the case $g_n \geq 2$*
- *a quotient $\mathtt{C}/\Gamma$ where Γ is a discrete subgroup of $\mathtt{C}$, in the case $g_n = 1$*
- *the Riemann sphere S^2, in the case $g_n = 0$*

By considering the possible discrete subgroups of $\mathbb{C}$, we obtain the following,

Corollary 2.1.2. *If Σ is a compact Riemann surface of genus 1, then it is diffeomorphic to the quotient of $\mathbb{C}$ by Γ, where Γ is lattice generated by 1 and τ, such that,*

- $Im\,\tau > 0$
- $\frac{1}{2} < Re\,\tau < \frac{1}{2}$
- $|\tau| > 1$
- $Re\,\tau \geq 0$ *if* $|\tau| = 1$

We will have need of the following theorem,

Theorem 2.1.3. *Suppose Σ is a compact Riemann surface with homology basis $A_1, B_1, ..., A_{n_g}, B_{n_g}$, $p_1, ..., p_n \in \Sigma$, and $r_1, ..., r_n \in \mathbb{C}$ with $\sum_k r_k = 0$. Then there exists a unique meromorphic one-form α on Σ such that α has vanishing A-periods and is holomorphic everywhere except at p_k where it has a simple pole of residue r_k.*

The case where the prescribed residues are all 0 is a special case, and we have that the mapping $\omega \mapsto \left(\int_{A_1} \omega, ..., \int_{A_{n_g}} \omega\right)$ is an isomorphism between holomorphic one-forms on Σ and $\mathbb{C}^{n_g}$. Thus we can find a unique cohomology basis $\alpha_1, ..., \alpha_{n_g}$ such that,

$$\int_{A_i} \alpha_j = \delta_{i,j}$$

The vectors $P_j = \left(\int_{A_1} \alpha_j, ..., \int_{A_{n_g}} \alpha_j\right)$ and $P_{n_g+j} = \left(\int_{B_1} \alpha_j, ..., \int_{B_{n_g}} \alpha_j\right)$ are called periods of Σ. Letting Λ consist of the lattice spanned by the period vectors, we have the following,

Theorem 2.1.4. (Abel) *Suppose Σ is a compact Riemann surface with $z_1, ...z_n$, $w_1, ...w_n \in \Sigma$, not necessarily distinct. There exists a meromorphic function on Σ with an mth order zero at each z_k and an pth order pole at each w_k, if and only if,*

$$\left(\sum_k \int_{w_k}^{z_k} \alpha_1, ..., \sum_k \int_{w_k}^{z_k} \alpha_{n_g} \right) \equiv 0 \quad (\mod \Lambda)$$

On a torus we have a single one-form α_1, which corresponds to the flat one-form dz in standard coordinates on $\mathbb{C}$ on a uniformized torus with basis vector 1. Thus we have the following corollary,

Corollary 2.1.5. *For a torus Σ, with w_k and z_k represented in coordinates, there exists a meromorphic function with the prescribed zeroes and poles (as before) if and only if,*

$$\sum_k w_k - z_k \in \Gamma$$

That is, the sum of vectors from zeroes to corresponding poles must be an element of the defining lattice of the torus.

2.2 Minimal Surfaces

If X is the coordinate function of an immersed minimal surface, then it inherits a natural conformal structure from $\mathbb{E}^3$, and on this Riemann surface the coordinate functions are harmonic. With the Hodge star operator, we can thus construct holomorphic one-forms $\Phi_k = dX_k + i \star dX_k$. The form Φ_3 is often named dh, despite its not being globally exact. We follow Traizet and will refer to it as η. The function $g = -\frac{\Phi_1 + i\Phi_2}{\Phi_3}$ recovers the stereographic projection of the Gauss map of the surface.

We can recover the original surface X, up to translation, from the Weierstrass

data g and η, using the equation,

$$X(p) = Re \int^p (\Phi_1, \Phi_2, \Phi_3)$$

Noting that we can recover,

$$\Phi_1 = \frac{1}{2}(\frac{1}{g} - g)\eta$$
$$\Phi_2 = \frac{i}{2}(\frac{1}{g} + g)\eta$$
$$\Phi_3 = \eta$$

we arrive at the Weierstrass representation. This presentation of the global Weierstrass representation theorem is due to Osserman (see [Osserman86]):

Theorem 2.2.1. (Weierstrass representation) *Let Σ be a Riemann surface, and suppose that η is meromorphic one-form and g is a non-constant meromorphic function on Σ, such that each zero of η of order n is either a pole or zero of g of order n, and vice versa. Then, with the following definitions, X, if well-defined, is a minimal immersion with ends at the poles of η,*

$$\Phi_1 = \frac{1}{2}(\frac{1}{g} - g)\eta$$
$$\Phi_2 = \frac{i}{2}(\frac{1}{g} + g)\eta$$
$$\Phi_3 = \eta$$
$$X(p) = Re \int_*^p (\Phi_1, \Phi_2, \Phi_3)$$

Here the * in the limits of integration indicates that the choice of basepoint doesn't

matter; it only changes X by a translation. The well-definedness of X in this equation is known as the period problem, and is equivalent to the condition that, for any closed loop γ,

$$Re \int_{\gamma} \Phi_k = 0$$

A simply periodic minimal surface such as Scherk's second surface has infinite genus in $\mathbb{R}^3$, but it is convenient to view it as a surface of finite topology in the quotient space $\mathbb{R}^3/(0,0,1)$. As this quotient space is flat, all of the local arguments go through. We have the following version of the Weierstrass representation theorem for simply periodic minimal surfaces, due to Meeks and Rosenberg [Meeks93]:

Theorem 2.2.2. (Weierstrass representation of simply-periodic surface)

Let Σ be a Riemann surface, and suppose that η is meromorphic one-form and g is a non-constant meromorphic function on Σ, such that each zero of η of order n is either a pole or zero of g of order n, and vice versa. Then, with the following definitions, $X : \Sigma \to \mathbb{E}^3/(0,0,1)$, if well-defined, is a minimal immersion with ends at the poles of η,

$$\begin{aligned}
\Phi_1 &= \frac{1}{2}(\frac{1}{g} - g)\eta \\
\Phi_2 &= \frac{i}{2}(\frac{1}{g} + g)\eta \\
\Phi_3 &= \eta \\
X(p) &= Re \int_{*}^{p} (\Phi_1, \Phi_2, \Phi_3)
\end{aligned}$$

The period problem in the quotient space is,

$$Re \int_\gamma \Phi_1 = 0$$
$$Re \int_\gamma \Phi_2 = 0$$
$$Re \int_\gamma \Phi_3 = 0 \pmod 1$$

2.3 Teichmuller theory

Let Σ be a closed Riemann surface of genus n_g (the base surface) and punctures $p_1, ..., p_n$, and consider pairs (Σ', f) where Σ' is another Riemann surface of genus n_g, and $f : \Sigma \to \Sigma'$ is an orientation-preserving diffeomorphism from the base surface. We define two such pairs (Σ'_1, f_1) and (Σ'_2, f_2) to be equivalent if $f_2 \circ f_1^{-1}$ is homotopic (respecting punctures) to a biholomorphism. The set of all equivalence classes is called the Teichmüller space of Σ, and will be denoted $\mathrm{T}(\Sigma)$.

Theorem 2.3.1. *If Σ is a closed Riemann surface with n punctures and genus $n_g \geq 1$, then $\mathrm{T}(\Sigma)$ is a smooth $6n_g - 6 + 2n$ real-dimensional manifold.*

A Beltrami differential is a type (1,-1) differential, ie an object that transforms like $\mu(z)\frac{d\bar{z}}{dz}$. The most important result concerning these objects is,

Theorem 2.3.2. *Given a Beltrami differential μ on a Riemann surface Σ such that $\|\mu\|_\infty < 1$, there exists a Riemann surface Σ' and a map $f : \Sigma \to \Sigma'$ that solves the Beltrami equation in a coordinate system:*

$$f_{\bar{z}}(z) = \mu(z) f_z(z) \tag{2.1}$$

Conversely, if we started with a Σ' and f, then (2.1) can serve as the definition of

μ from which the preceding theorem lets us recover Σ' up to conformal isomorphism. Thus Beltrami differentials can be used to parameterize $\mathrm{T}(\Sigma)$ up to equivalence. Another important class of objects on a Riemann surface are the holomorphic quadratic differentials, which are tensors that transform like $f(z)dz^2$ for f some locally holomorphic function. As the exterior product of a dz form and a $d\overline{z}/dz$ form produces a multiple of the area form, there is a natural pairing between bounded Beltrami differentials and integrable holomorphic quadratic differentials on Σ, given by,

$$(\Phi, \mu) = \int_{\Sigma} \Phi\mu$$

We have the following characterization of the infintessimal structure of Teichmüller space:

Theorem 2.3.3. *The cotangent space* $\mathrm{T}^*(\Sigma)$ *can be naturally identified with the space of integrable holomorphic quadratic differentials on* Σ*, and the tangent space* $\mathrm{T}_*(\Sigma)$ *with equivalence classes of bounded measurable Beltrami differentials on* Σ *under the equivalence defined by* $\mu_1 \sim \mu_2$ *if* $(\Phi, \mu_1 - \mu_2) = 0$ *for all integrable holomorphic quadratic differentials* Φ *on* Σ.

In the preceding, the requirement is that the quadratic differentials be holomorphic on the Riemann surface Σ with punctures removed. Viewed as a compact Riemann surface, we may thus find co-vectors that are properly meromorphic. However, the requirement that they be integrable restricts them to including only simple poles.

Chapter 3

The Constraints and Their Independence

3.1 Symmetric Riemann surfaces

We assume we are given Σ, a Riemann surface of genus n_g with a set of punctures labelled z_1^+, ..., $z_{n_z}^+$, z_1^-, ..., $z_{n_z}^-$, and e_1^+, ..., $e_{n_e}^+$, e_1^-, ..., $e_{n_e}^-$ and toplogy basis $A_1, ..A_{n_p}$ and $B_1, ..B_{n_p}$ with common basepoint b (distinct from the punctures) such that the punctures e_i^+ and e_i^- lie on B_1. We further assume that we are given σ, an anti-conformal involution of Σ such that:

$$\begin{aligned} \sigma(z_i^+) &= z_i^- \qquad \sigma(z_i^-) = z_i^+ \\ \sigma(e_i^+) &= e_i^+ \qquad \sigma(e_i^-) = e_i^- \\ \sigma(A_i) &= -A_i \qquad \sigma(B_i) = B_i \end{aligned}$$

The e_i^+ and e_i^- are to be the ends of our surface. The z_i^+ and z_i^- are to be the zeroes and poles of g. See Figure 4.14 in chapter 4.

As we wish to study surfaces admiting a reflective symmetry, we will require an understanding of Riemann surfaces that admit an anticonformal involution. We let σ denote an anticonformal involution of a Riemann surface Σ.

For any differential ω, we introduce the notation:

$$\omega^+ = \frac{1}{2}(\omega + \overline{\sigma^*\omega})$$
$$\omega^- = \frac{1}{2}(\omega - \overline{\sigma^*\omega})$$

Consider the map $\omega \mapsto \overline{\sigma^*\omega}$. If Ω is any subspace of differentials closed under this map, it is easily checked that for any $\omega \in \Omega$ we have that ω^+ and ω^- are eigenvectors of the map corresponding to eigenvalues 1 and -1, respectively. Since $\omega = \omega^+ + \omega^-$ for any ω, we have a decomposition of differentials intc respective eigenspaces Ω^+ and Ω^-.

Definition 3.1.1. We will call any differential in Ω^+ *symmetric* and any differential in Ω^- *anti-symmetric.*

For later use, we collect some simple facts.

$$\omega^+ = \overline{\sigma^*\omega^+} \tag{3.1}$$

$$\omega^- = -\overline{\sigma^*\omega^-} \tag{3.2}$$

For any curve γ and one-form ω, we have:

$$\int_\gamma \omega = \int_{\sigma(\gamma)} \sigma^*\omega = \overline{\int_{\sigma(\gamma)} \overline{\sigma^*\omega}}$$

and thus we have that:

$$\mathrm{Res}|_p\omega = -\overline{\mathrm{Res}|_{\sigma(p)}\overline{\sigma^*\omega}}$$

$$\int_{A_i} \omega = -\overline{\int_{A_i} \overline{\sigma^*\omega}}$$

$$\int_{B_i} \omega = \overline{\int_{B_i} \overline{\sigma^*\omega}}$$

It follows that a characterization of symmetric and anti-symmetric one-forms is

Proposition 3.1.2. *A one-form ω is symmetric if and only if:*

$$Im\mathrm{Res}|_{e_i^+}\omega = Im\mathrm{Res}|_{e_i^-}\omega = 0$$

$$\mathrm{Res}|_{z_i^-}\omega = \overline{Res|_{z_i^+}\omega}$$

$$Re\int_{A_i} \omega = 0$$

A one-form ω is antisymmetric if and only if:

$$Re\mathrm{Res}|_{e_i^+}\omega = Re\mathrm{Res}|_{e_i^-}\omega = 0$$

$$\mathrm{Res}|_{z_i^-}\omega = -\overline{\mathrm{Res}|_{z_i^+}\omega}$$

$$Im\int_{A_i} \omega = 0$$

We also note that, due to (3.1) and (3.2), the product of two symmetric dif-

ferentials or two anti-symmetric differentials is symmetric, while the product of a symmetric and anti-symmetric differential is anti-symmetric.

We now turn to some Teichmüller theory. Given a Beltrami differential μ on Σ, with $\|\mu\| < 1$, we may solve the Beltrami equation to produce a Reimann surface Σ^μ with conformal map $f^\mu : \Sigma \to \Sigma^\mu$. We would like to consider only labelled Riemann surfaces Σ^μ that deform Σ symmetrically, such that we can lift σ to each Σ^μ in a natural way. We define,

$$\sigma^\mu = f^\mu \circ \sigma \circ (f^\mu)^{-1} \tag{3.3}$$

We check that $\sigma^\mu \circ \sigma^\mu = id$, so it is an involution, and we seek to investigate under what conditions σ^μ is anticonformal.

Proposition 3.1.3. *Equation (3.3) defines an anti-conformal involution σ^μ if and only if μ is symmetric, or equivalently, if and only if,*

$$\overline{\sigma^* \mu} = \mu \tag{3.4}$$

Proof. We rewrite (3.3) as,

$$\sigma^\mu \circ f^\mu = f^\mu \circ \sigma \tag{3.5}$$

We fix a point p and choose charts z defined in a neighborhood of p, z' defined in a neighborhood of $\sigma(p)$, w defined in a neighborhood of $f^\mu(p)$, and w' defined in a neighborhood of $(\sigma^\mu \circ f^\mu)(p) = (f^\mu \circ \sigma)(p)$. See Figure 3.1.

Using the fact that σ is anti-conformal ($\sigma_z = 0$), we differiante (3.5) with respect

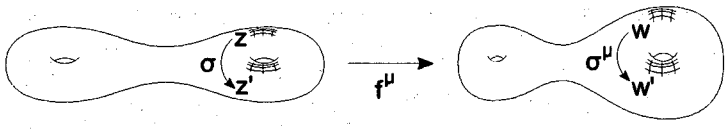

Figure 3.1 : Symmetry condition for a deformation

to z and $\overline{z}$, to obtain:

$$\sigma^{\mu}_{w} f^{\mu}_{z} + \sigma^{\mu}_{\overline{w}} (\overline{f^{\mu}})_{z} = f^{\mu}_{\overline{z'}} (\overline{\sigma})_{z} \tag{3.6}$$

$$\sigma^{\mu}_{w} f^{\mu}_{\overline{z}} + \sigma^{\mu}_{\overline{w}} (\overline{f^{\mu}})_{\overline{z}} = f^{\mu}_{z'} \sigma_{\overline{z}} \tag{3.7}$$

We solve for σ^{μ}_{w} by multiplying (3.6) by $(\overline{f^{\mu}})_{\overline{z}}$ and (3.7) by $-(\overline{f^{\mu}})_{z}$, adding the results:

$$\sigma^{\mu}_{w} = \frac{f^{\mu}_{\overline{z'}} (\overline{\sigma})_{z} (\overline{f^{\mu}})_{\overline{z}} - f^{\mu}_{z'} \sigma_{\overline{z}} (\overline{f^{\mu}})_{z}}{f^{\mu}_{z} (\overline{f^{\mu}})_{z} - f^{\mu}_{\overline{z}} (\overline{f^{\mu}})_{z}} \tag{3.8}$$

The denominator $|f^{\mu}_{z}|^2 - |f^{\mu}_{\overline{z}}|^2$ is non-vanishing since $\|\mu\|_{\infty} < 1$. We can therefore

rewrite the condition $\sigma^{\mu}_{w} = 0$ as:

$$\frac{f^{\mu}_{\overline{z}'}}{f^{\mu}_{z'}} = \frac{\overline{\sigma_{\overline{z}}}\,\overline{f^{\mu}_{\overline{z}}}}{\sigma_{\overline{z}}\,\overline{\overline{f^{\mu}_{z}}}} \tag{3.9}$$

We obtain a coordiante invariant form of this expression by computing (again using that σ is anti-conformal):

$$\overline{\sigma^{*}\mu} = \sigma^{*}\overline{(\frac{f^{\mu}_{\overline{z}'}d\overline{z'}}{f^{\mu}_{z'}dz'})} = \overline{(\frac{f^{\mu}_{\overline{z}'}\overline{\sigma_{\overline{z}}}dz}{f^{\mu}_{z'}\sigma_{\overline{z}}d\overline{z}})} = \frac{f^{\mu}_{\overline{z}}d\overline{z}}{f^{\mu}_{z}dz} = \mu \tag{3.10}$$

□

We now seek to find a characterization of the infinitesimal Beltrami differentials, considered as tangent vectors to the Teichmüller space $\mathrm{T}(\Sigma)$, in terms of their pairing with holomorphic quadratic differentials, covectors to $\mathrm{T}(\Sigma)$.

Proposition 3.1.4. *A holomorphic quadratic differential Φ is symmetric if and only if $(\Phi, \mu) = 0$ for all symmetric Beltrami differentials μ, and is antisymmetric if and only if $(\Phi, \mu) = 0$ for all antisymmetric Beltrami differentials μ.*

Proof. Remembering that σ is orientation-reversing, we have for any holomorphic quadratic differential Φ (symmetric or not),

$$\begin{aligned} 2(\Phi, \mu) &= \mathrm{Re}\int_{\Sigma}\Phi\mu + \mathrm{Re}\int_{\Sigma}\Phi\mu \\ &= \mathrm{Re}\int_{\Sigma}\Phi\mu + \mathrm{Re}\int_{\sigma(\Sigma)}\Phi\mu \\ &= \mathrm{Re}\int_{\Sigma}(\Phi\mu - \sigma^{*}(\Phi\mu)) \end{aligned}$$

If $\Phi = \Phi^+$ is symmetric, then for any symmetric $\mu = \mu^+$ we have,

$$\begin{aligned} 2(\Phi, \mu) &= \mathrm{Re} \int_\Sigma (\Phi\mu - \sigma^*(\Phi\mu)) \\ &= \mathrm{Re} \int_\Sigma (\Phi\mu - \sigma^*\Phi\,\sigma^*\mu) \\ &= \mathrm{Re} \int_\Sigma (\Phi\mu - \overline{\Phi\mu}) \\ &= 0 \end{aligned}$$

Conversely if Φ is not symmetric, then there exists some point p for which $\Phi(p) \neq \overline{\sigma^*(\Phi)}(p)$. Since Φ is continuous, we can choose a symmetric μ supported in a neighborhood of $\{p\} \cup \{\sigma(p)\}$ such that $(\Phi, \mu) \neq 0$.

Similarly, if $\Phi = \Phi^-$ is antisymmetric, then for any antisymmetric $\mu = \mu^-$ we have,

$$\begin{aligned} 2(\Phi, \mu) &= \mathrm{Re} \int_\Sigma (\Phi\mu - \sigma^*(\Phi\mu)) \\ &= \mathrm{Re} \int_\Sigma (\Phi\mu - \sigma^*\Phi\,\sigma^*\mu) \\ &= \mathrm{Re} \int_\Sigma (\Phi\mu - \overline{\Phi\mu}) \\ &= 0 \end{aligned}$$

Conversely if Φ is not antisymmetric, then there exists some point p for which $\Phi(p) \neq -\overline{\sigma^*(\Phi)}(p)$. Since Φ is continuous, we can choose a antisymmetric μ supported in a neighborhood of $\{p\} \cup \{\sigma(p)\}$ such that $(\Phi, \mu) \neq 0$. □

We now introduce $\mathrm{T}^\sigma(\Sigma)$, the symmetric Teichmüller space of (Σ, σ), as follows. We let $\sim$ denote the usual Teichmüller equivalence between Beltrami differentials and define $\mathrm{T}^\sigma(\Sigma)$ to be the quotient of symmetric Beltrami differentials by $\sim$. Let

$N = 3n_g + 2n_z + 2n_e - 3$ be half the number of real dimensions of $T(\Sigma)$. We let $\Phi_1^+, ...\Phi_N^+, \Phi_1^-, ..., \Phi_N^-$, be holomorphic quadratic differentials forming a basis for the real cotangent space to $T(\Sigma)$ at Σ, such that the Φ_i^+ are symmetric, and the Φ_i^- are antisymmetric. (Note: 'Real' here means that the pairing between tangent and cotangent vectors produces a real value, and we consider the space as a vector space over the reals.)

We can then choose a corresponding basis, $\chi_1^-, ..., \chi_N^-, \chi_1^+, ...\chi_N^+$, for the tangent space, such that

$$(\Phi_i^+, \chi_j^-) = \delta_{i,j} \qquad (\Phi_i^-, \chi_j^-) = 0$$
$$(\Phi_i^+, \chi_j^+) = 0 \qquad (\Phi_i^-, \chi_j^+) = \delta_{i,j}$$

We then define

$$\nu_i^- = (\chi_i^-)^-$$
$$\nu_i^+ = (\chi_i^+)^+$$

The ν^+ and ν^- are thus symmetric and anti-symmetric, and thus from Proposition 3.1.4, we know that,

$$(\Phi_i^+, \nu_j^-) = (\Phi_i^+, \chi_j^- - (\chi_j^-)^+) = (\Phi_i^+, \chi_j^-) = \delta_i, j$$
$$(\Phi_i^+, \nu_j^+) = 0$$
$$(\Phi_i^-, \nu_j^-) = 0$$
$$(\Phi_i^-, \nu_j^+) = (\Phi_i^-, \chi_j^+ - (\chi_j^+)^-) = (\Phi_i^-, \chi_j^+) = \delta_i, j$$

We introduce a local coordinate on $\mathrm{T}^\sigma(\Sigma)$ at Σ as follows. Define the map,

$$g : R^N \to \mathrm{T}(\Sigma)$$
$$g(x) = x \cdot (\nu_1^+, ..., \nu_N^+)$$

The image of this map is exactly the symmetric Beltrami differentials, and thus all $\mathrm{T}^\sigma(\Sigma)$, considered a subset of $\mathrm{T}(\Sigma)$ under inclusion. The image of the tangent space of this map at 0 is full rank, and thus by the inverse function theorem is locally 1-1. We can thus consider $\mathrm{T}^\sigma(\Sigma)$ to be, locally (which is all we are interested in), an $3n_g+2n_z+2n_e-3$ dimensional submanifold of $\mathrm{T}(\Sigma)$ with (real) tangent space spanned by equivalence classes of symmetric L^∞ Beltrami differentials and (real) cotangent space spanned by the antisymmetric holomorphic quadratic differentials.

3.2 Weierstrass data

Given a point on a certain sublocus of the symmetric Teichmüller space $\mathrm{T}(\Sigma, \sigma)$, we will associate Weierstrass data for a complete embedded simply-periodic minimal surface with Scherk type ends admitting a reflective symmetry in the direction of periodicity. The construction is essentially the one used in [Traizet00], though the formulation is slightly different because he performs his construction on a disjoint union of spheres with identified nodes.

Our goal will be to define η^μ and g^μ on Σ^μ such that a complete, embedded, symmetric minimal surface with Scherk type ends is produced by the Weierstrass representation:

$$\phi_1^\mu = \frac{1}{2}(\frac{1}{g^\mu} - g^\mu)\eta^\mu$$
$$\phi_2^\mu = \frac{i}{2}(\frac{1}{g^\mu} + g^\mu)\eta^\mu$$
$$\phi_3^\mu = \eta^\mu$$

$$X^\mu = Re \int (\phi_1, \phi_2, \phi_3)$$

Our construction will not apply to all μ, however; we will need to impose some constraints.

Next we let η^μ to be the unique one-form with poles at the e_i^+ and e_i^- such that,

$$Res|_{e_i^+} \eta^\mu = \frac{1}{2\pi i}$$
$$Res|_{e_i^-} \eta^\mu = \frac{-1}{2\pi i}$$

$$\int_{A_i} \eta^\mu = 1$$

Proposition 3.2.1. *The differential η^μ is anti-symmetric, or*

$$\eta^\mu = -\overline{\sigma^* \eta^\mu} \tag{3.11}$$

Proof. The residues of η^μ at the e_i^+ or e_i^- are imaginary. Also, the A_i-periods of η^μ are purely real. The result follows from Proposition 3.1.2. □

Next, we attempt to define a function g^μ. Supposing that we had the function g^μ

on Σ^μ, normalized so that $g(e_1^-) = 1$, we could define a one-form ψ^μ by,

$$\psi^\mu = d(\log g^\mu) = \frac{dg^\mu}{g^\mu}$$

from which we could recover g^μ by,

$$g^\mu(x) = e^{\int_{e_1^-}^{x} \psi^\mu} \tag{3.12}$$

We are thus led to define ψ^μ to be the unique holomorphic differential on Σ^μ with poles at the z_i^+ and z_i^- such that,

$$Res|_{z_i^+} \psi^\mu = 1$$

$$Res|_{z_i^-} \psi^\mu = -1$$

$$\int_{A_i} \psi^\mu = 0$$

We would like to define g^μ by (3.12), but it is only well defined if $\int_\gamma \psi^\mu$ is a multiple of $2\pi i$ for each closed curve γ on Σ. From the definition of ψ_μ, we know this condition holds for A_i-curves or loops around poles, so it suffices to establish the condition for the B_i-curves.

Proposition 3.2.2. *The differential ψ^μ is anti-symmetric, or*

$$\psi^\mu = -\overline{\sigma^* \eta^\mu} \tag{3.13}$$

Proof. The residues of ψ^μ at the z_i^+ or z_i^- are real. Also, the A_i-periods of ψ^μ are purely real. The result follows from Proposition 3.1.2. □

It follows from this proposition that the B_i-periods of ψ^μ are purely imaginary. The first kind of constraint that we wish to impose on μ is that, for each curve B_i,

$$\text{Condition 1:} \qquad \operatorname{Im} \int_{B_i} \psi^\mu = 0 \pmod{2\pi}$$

We also note the following fact, which we will make use of later:

Proposition 3.2.3. *When it exists, g^μ satisfies*

$$g^\mu = \frac{1}{\overline{\sigma^* g^\mu}}$$

Proof. Apply the previous proposition to the definition of g^μ in (3.12). □

In order to ensure that our surface has Scherk-type ends, we require that the set zeroes of η is equal to the set of zeroes and poles of g. This is our second kind of constraint,

$$\text{Condition 2:} \qquad \eta^\mu(z_i^+) = 0$$

$$\text{Condition 3:} \qquad \eta^\mu(z_i^-) = 0$$

Finally, we wish to ensure that our surface solves the so-called "period problem".

We must show that for any curve γ on Σ^μ, we have,

$$\begin{aligned}
\mathrm{Re}\int_\gamma \phi_1^\mu &= 0 \\
\mathrm{Re}\int_\gamma \phi_2^\mu &= 0 \\
\mathrm{Re}\int_\gamma \phi_3^\mu &= 0 \pmod 1
\end{aligned}$$

For $\phi_3^\mu = \eta^\mu$, we know that the A-period is 1 and that for any loop γ around a single pole of η^μ,

$$\int_\gamma \phi_3^\mu = \pm 1$$

Since η is anti-symmetric, we also have that,

$$\mathrm{Re}\int_{B_i} \phi_3^\mu = 0$$

Following Traizet, we compute that for any curve γ with $\sigma(\gamma) = \gamma$,

$$\begin{aligned}
\int_\gamma \phi_1^\mu &= \int_\gamma \frac{1}{2}(\frac{1}{g^\mu} - g^\mu)\eta^\mu \\
&= \int_{\sigma(\gamma)} \sigma^*(\frac{1}{2}(\frac{1}{g^\mu} - g^\mu)\eta^\mu) \\
&= \int_{-\gamma} \frac{1}{2}(\overline{g}^\mu - \frac{1}{\overline{g}^\mu})(-\overline{\eta}^\mu) \\
&= -\overline{\int_\gamma \frac{1}{2}(\frac{1}{g^\mu} - g^\mu)\eta^\mu} \\
&= -\overline{\int_\gamma \phi_1^\mu}
\end{aligned}$$

Similarly we compute,

$$\begin{aligned}\int_\gamma \phi_2^\mu &= \int_\gamma \frac{i}{2}(\frac{1}{g^\mu} + g^\mu)\eta^\mu \\ &= \int_{\sigma(\gamma)} \sigma^*(\frac{i}{2}(\frac{1}{g^\mu} + g^\mu)\eta^\mu) \\ &= \int_{-\gamma} \frac{i}{2}(\overline{g}^\mu + \frac{1}{\overline{g}^\mu})(-\overline{\eta}^\mu) \\ &= -\overline{\int_\gamma \frac{i}{2}(\frac{1}{g^\mu} + g^\mu)\eta^\mu} \\ &= -\overline{\int_\gamma \phi_2^\mu}\end{aligned}$$

We conclude that,

$$\mathrm{Re}\int_{\gamma_i} \phi_1^\mu = 0$$

$$\mathrm{Re}\int_{\gamma_i} \phi_2^\mu = 0$$

Thus the A_i-periods of ϕ_1^μ and ϕ_2^μ are purely imaginary, and the residue at the e_i^+ and e_i^- (which are the only poles) are real. If we also knew that the B_i-periods of ϕ_1^μ and ϕ_2^μ vanished, then we would have solved the period problem for our surface. We are thus led to our final constraints,

$$\begin{aligned}&\text{Condition 4:} \quad \mathrm{Re}\int_{B_i} \phi_1^\mu = 0 \\ &\text{Condition 5:} \quad \mathrm{Re}\int_{B_i} \phi_2^\mu = 0\end{aligned}$$

We summarize the discussion as,

Proposition 3.2.4. *For each point $\Sigma^\mu \in \mathrm{T}^\sigma(\Sigma)$ the Weierstrass data (g^μ, η^μ) produces a complete simply-periodic minimal surfaces with Scherk-type ends with a reflective symmetry in the direction of periodicity if the following conditions are met:*

$$Im \int_{B_i} \psi^\mu = 0 \quad (mod\, 2\pi)$$
$$\eta^\mu(z_i^+) = 0$$
$$\eta^\mu(z_i^-) = 0$$
$$Re \int_{B_i} \phi_1^\mu = 0$$
$$Re \int_{B_i} \phi_2^\mu = 0$$

These equations define G, H, I, J, and K, as

Proof. Collect conditions 1 through 5 above. □

We note the following,

Proposition 3.2.5. *If the surface defined at $\mu = 0$ is embedded and has no pair of ends that are asymptopically parallel, then in some neighborhood of $\mu = 0$, every surface satisfying conditions 1-5 is embedded.*

Proof. First, we may choose a neighborhood S_1 of $\mu = 0$ such that the curvature of the surface is bounded on the neighborhood. With bounded curvature, we cannot have local intersections. That is, for every $p \in \Sigma$ we can choose some neighborhood O_p of p such that $q \in O_p$ implies $X^\mu(p) \neq X^\mu(q)$ for all $\mu \in S_1$. If no pair of ends is asymptopically parallel, we can choose a compact domain $K \subset \Sigma$ and a neighborhood S_2 of $\mu = 0$ such that the complement of K consists of sets E_k corresponding to the ends of the surface such that each $X^\mu(E_k)$ is a graph over its corresponding end and such that, by restricting S_2 if needed, the images $X^\mu(E_k)$ are disjoint for each $\mu \in S_2$.

As each end is a graph, we can further restrict S_2 such that $X^\mu(E_k)$ and $X(K)$ are disjoint. Now, for $p \in K$, define $d^\mu(p)$ to be $min_{q \in K - O_p} dist(X^\mu(p), X^\mu(q))$. This is a continuous function on a compact set, bounded above zero for $\mu = 0$ since our base surface is embedded, and hence can choose some neighborhood S_3 of $\mu = 0$ on which it is bounded above zero. Thus, on the neighborhood $S_1 \cap S_2 \cap S_3$, we know that ends don't intersect themselves, since they are graphs. We know that ends don't intersect other ends or $X^\mu(K)$. We know that for distinct $p, q \in K$, we have $X^\mu(p) \neq X^\mu(q)$, since either $q \in O_p$ and the curvature bound applies, or $p \in K - O_p$ and the bound on d^μ applies, thus $X^\mu(K)$ does not intersect itself. This rules out all possible intersections. □

We will apply the results of this section by starting from a set of data (Σ, σ) for which the constructions in this section are known to produdce an embedded surface satisfying all of the constraints of Proposition 3.2.4. Examples of such surfaces include those in Karcher's family as well as the examples produced by Traizet in [Traizet00]. We will next develop functions encoding the constraints on Teichmüller space and then apply the implicit function theorem to satisfy them in a neighborhood of the original.

As a convenience, we will drop the 0 superscript for μ when referring to functions and forms on the base surface. For example, we will refer to ψ^0 as ψ, g^0 as g, and η^0 as η.

3.3 Rauch calculus

In the previous section, the forms η^μ and ψ^μ were constructed by specifying their residues and A_i-periods. To understand how forms defined in this manner vary with μ, we develop a Rauch variational formula. We proceed exactly as in [Earle], with

only minor modifications to allow for the case of meromorphic forms.

A given one-form ω can be written uniquely as $\omega = \omega' + \omega''$ with ω' a type (1,0) form and ω'' a type (0,1) form on Σ. If ω is holomorphic on Σ^μ, and w is a local conformal coordinate on Σ_μ, then ω is locally of the form $f(w)dw$. On the other hand, $w = w(z)$ must satisfy

$$w_{\overline{z}} = \mu w_z$$

Thus, in terms of z, the local form of ω is

$$f(w(z))w_z(z)(dz + \mu(z)d\overline{z})$$

Thus $\omega' = f(w(z))w_z(z)dz$ and $\omega'' = f(w(z))w_z(z)\mu(z)d\overline{z}$. Hence,

$$\omega'' = \mu\omega' \tag{3.14}$$

Conversely, any ω satisfying (3.14) is holomorphic on Σ^μ.

Lemma 3.3.1. *For a type-*$(0,1)$ *form* ω *on* Σ *there is a unique form* $L\omega$ *of type-*$(1,0)$ *on* Σ *such that:*

$$d(\omega + L\omega) = 0$$
$$\int_{A_i} \omega + L\omega = 0$$

Proof. If σ_1 and σ_2 both satisfy the conditions of the lemma, then $\sigma_1 - \sigma_2$ is holomorphic and has zero A-periods, and is thus zero. To prove existence, write $\omega = \overline{\alpha} + \overline{\partial}f$, where α is a holomorphic one-form and f is a C^∞ function. Then $L\omega = \beta + \partial f$, where

β, holomorphic, is chosen such that,

$$\int_{A_j} \beta = -\int_{A_j} \overline{L\omega}$$

Then $L\omega$ has the right type, the A-periods of $\beta + \overline{\alpha}$, ∂f, and $\overline{\partial} f$ vanish, and $\overline{\alpha}$, β, and $\partial f + \overline{\partial} f$ are closed. □

Now, fix ω to be a meromorphic form on Σ and let ω_μ be the unique one-form on Σ^μ with the same A_i-periods as ω and the same residues as ω at corresponding poles. Then $\delta = \omega_\mu - \omega$ is a closed, *regular* one-form on Σ with vanishing A_i-periods, satisfying,

$$\delta'' = \omega''_\mu$$
$$\delta' = \omega'_\mu - \omega$$

It follows that,

$$L(\delta'') = \delta'$$

Then we have,

$$\mu\delta' = \mu(\omega'_\mu - \omega) = \delta'' - \mu\omega$$

Thus,

$$L(\mu\delta' + \mu\omega) = L(\delta'') = \delta'$$

and,

$$(I - L\mu)\delta' = L\mu\omega$$

Note that because μ is supported away from the poles of ω, we have that $\mu\omega$ is a type-$(0,1)$ form on Σ. We then invert $(I - L\mu)$ to solve for δ', exactly as in [Earle]. We thus obtain the formula:

$$\delta' = \sum_{n=0}^{\infty} (L\mu)^n (L\mu\omega)$$

Thus,

$$(\omega^\mu)' = \omega + \sum_{n=1}^{\infty} (L\mu)^n \omega = \omega + o(\|\mu\|)$$
$$(\omega^\mu)'' = \mu(\omega^\mu)' = \mu\omega + o(\|\mu\|^2)$$

Proposition 3.3.2. *If α is a meromorphic form on Σ and β^μ is a meromorphic form on Σ^μ (with $\beta = \beta^0$), then,*

$$\alpha \wedge \beta^\mu = \alpha\beta\mu + o(\|\mu\|^2)$$

Proof.

$$\begin{aligned} \alpha \wedge \beta^\mu &= \alpha \wedge (\beta^{\mu'} + \beta^{\mu''}) \\ &= \alpha \wedge \beta^{\mu''} \\ &= \alpha \wedge (\mu\beta + o(\|\mu\|^2)) \\ &= (\alpha\beta)\mu + o(\|\mu\|^2) \end{aligned}$$

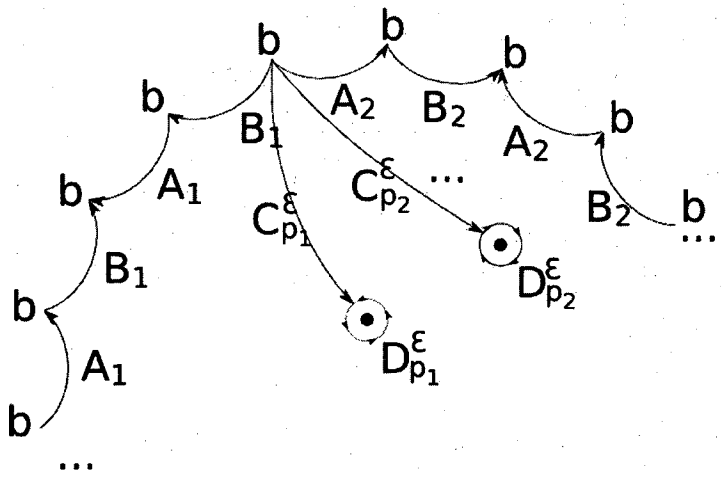

Figure 3.2 : Fundamental domain for bilinear relations

□

3.4 Bilinear relations

In this section we develop the bilinear relations in a form convenient for what follows. In some sense it is more of a bilinear relations kit that can be finished out in the context where it is needed.

We recall that b is the basepoint at the common intersection of the A_i and B_i.

Definition 3.4.1. For each puncture p, choose a curve C_p from b to p in such a way that

1. None of the curves C_p intersect each other or the A_i or B_i curves except at b.

2. For sufficiently small $\epsilon > 0$ the curve only intersects the ϵ-circle centered at q in a single point.

We let C_p^ϵ to be the initial part of the path C_p from b to the point of intersection of C_p and the ϵ-circle. We let D_i^ϵ to be a clockwise loop around the ϵ-circle beginning and ending at the endpoint of C_p^ϵ.

Removing from Σ the ϵ-disks centered at each puncture p and the curves A_i, B_i, C_p, we have an open simply connected region Σ^ϵ. We form a closure of this region under the subspace topology that treats points separated by A_i, B_i, and C_p^ϵ as far apart. The result is that each point on the curves A_i, B_i, and C_p^ϵ is duplicated, being traversed once forward and once backward during a clockwise walk around the boundary. We label the new curves A_i^+, B_i^+, and $C_p^{\epsilon+}$ when traversed forward and A_i^-, B_i^-, and $C_p^{\epsilon-}$ when traversed backwards. See Figure 3.2. This procedure applies equally well when $\epsilon = 0$, resulting in Σ^0.

Proposition 3.4.2. *Suppose that α, β are closed one-forms on Σ. Then there exists f such that $\alpha = df$ on the interior of Σ^0, and*

$$\int_\Sigma \alpha \wedge \beta = \sum_i \left(\int_{A_i} \alpha \int_{B_i} \beta - \int_{A_i} \beta \int_{B_i} \alpha \right) + \lim_{\epsilon \to 0} \sum_p \int_{C_p^\epsilon D_p^\epsilon C_p^{\epsilon -1}} f\beta$$

Proof. A closed form on a simply connected domain is exact, and we can thus write $\alpha = df$ for some function f on Σ^ϵ. Since $f - f(b)$ also satisfies $\alpha = df$, we will assume

that $f(b) = 0$. Since β is closed, integration by parts yields:

$$\begin{aligned}
\int_{\Sigma} \alpha \wedge \beta &= \lim_{\epsilon \to 0} \int_{\Sigma^\epsilon} \alpha \wedge \beta \\
&= \lim_{\epsilon \to 0} \int_{\Sigma^\epsilon} df \wedge \beta \\
&= \lim_{\epsilon \to 0} \left(\int_{\Sigma^\epsilon} d(f\beta) - \int_{\Sigma^\epsilon} f \wedge d\beta \right) \\
&= \lim_{\epsilon \to 0} \int_{\partial \Sigma^\epsilon} f\beta \\
&= \sum_i \int_{A_i B_i A_i^{-1} B_i^{-1}} f\beta + \lim_{\epsilon \to 0} \sum_p \int_{C_p^\epsilon D_p^\epsilon C_p^{\epsilon\,-1}} f\beta
\end{aligned}$$

To simplify the first term, note that if x^+ and x^- are points on A_i^+ and A_i^-, respectively, that correspond to the same point on Σ^ϵ, then we have:

$$f(x^-) = f(x^+) + \int_{B_i} df$$

Similarly, if x^+ and x^- are points on B_i^+ and B_i^- that correspond to the same point on Σ^ϵ, then we have:

$$f(x^-) = f(x^+) - \int_{A_i} df$$

Thus,

$$\begin{aligned}\int_{A_iB_iA_i^{-1}B_i^{-1}} f\beta &= \int_{A_iA_i^{-1}} f\beta + \int_{B_iB_i^{-1}} f\beta \\ &= \int_{A_i} f(x^+) - f(x^-) + \int_{B_i} f(x^+) - f(x^-) \\ &= \int_{A_i} (\int_{B_i} -df)\beta + \int_{B_i} (\int_{A_i} df)\beta \\ &= \int_{A_i} \alpha \int_{B_i} \beta - \int_{A_i} \beta \int_{B_i} \alpha\end{aligned}$$

□

Proposition 3.4.3. *If at puncture p, α is regular, and β has a first order pole, then we have:*

$$\lim_{\epsilon \to 0} \int_{C_p^\epsilon D_p^\epsilon C_p^{\epsilon^{-1}}} f\beta = 2\pi i \mathrm{Res}|_p \beta \int_{C_p} \alpha$$

Proof. If x^+ and x^- are corresponding points on C_p^ϵ and $C_p^{\epsilon^{-1}}$, then,

$$f(x^-) = f(x^+) + \int_{D_p^\epsilon} df = f(x^+) + \int_{D_p^\epsilon} \alpha = f(x^+)$$

Hence,

$$\int_{C_p^\epsilon C_p^{\epsilon^{-1}}} f\beta = 0$$

Then, using our choice of $f(b) = 0$,

$$\begin{aligned}\lim_{\epsilon\to 0}\int_{D_p^\epsilon} f\beta &= 2\pi i f(p)\mathrm{Res}|_p\beta \\ &= 2\pi i\left(f(b) + \int_{C_p}\alpha\right)\mathrm{Res}|_p\beta \\ &= 2\pi i\mathrm{Res}|_p\beta\int_{C_p}\alpha\end{aligned}$$

Combining terms gives the result. □

Proposition 3.4.4. *If at puncture p, α has a first order pole, and β is regular, then we have:*

$$\lim_{\epsilon\to 0}\int_{C_p^\epsilon D_p^\epsilon C_p^{\epsilon^{-1}}} f\beta = -2\pi i\mathrm{Res}|_p\alpha\int_{C_p}\beta$$

Proof. If x^+ and x^- are corresponding points on C_i^ϵ and $C_i^{\epsilon^{-1}}$, then,

$$f(x^-) = f(x^+) + \int_{D_p^\epsilon} df$$

Thus,

$$\begin{aligned}\lim_{\epsilon\to 0}\int_{C_p^\epsilon C_p^{\epsilon^{-1}}} f\beta &= \lim_{\epsilon\to 0}\int_{C_p^\epsilon}(f(x^+) - f(x^-))\beta \\ &= \lim_{\epsilon\to 0}\int_{C_p^\epsilon}(\int_{D_p^\epsilon} -df)\beta \\ &= -\lim_{\epsilon\to 0}\int_{C_p^\epsilon}\beta\int_{D_p^\epsilon}\alpha \\ &= -2\pi i p\mathrm{Res}|_\alpha\int_{C_p}\beta\end{aligned}$$

To evaluate the term $\int_{D_p^\epsilon} f\beta$, we note that, since $df = \alpha$ has a simple pole and β is

regular, $|f| < c_1 \log|\epsilon|$ and $\beta < c_2$ for some constants c_1 and c_2. Thus,

$$|\int_{D_p^\epsilon} f\beta| < 2\pi c_1 c_2 |\epsilon| \log|\epsilon| \to 0 \quad \text{as } |\epsilon| \to 0$$

Combining terms gives the result. □

Corollary 3.4.5. *If at puncture p, α and β are regular, then:*

$$\lim_{\epsilon \to 0} \int_{C_p^\epsilon D_p^\epsilon C_p^{\epsilon -1}} f\beta = 0$$

3.5 Continued existence of g

We now attempt to satisfy the first constraint of Proposition 3.2.4, namely,

$$\mathrm{Im} \int_{B_i} \psi^\mu = 0 \pmod{2\pi}$$

We define the function $\mathrm{G}_j : \mathrm{T}(\Sigma) \to R$ by,

$$\mathrm{G}_j(\Sigma^\mu) = -\mathrm{Im} \int_{B_j} \psi^\mu = \mathrm{Re} \int_{B_j} i\psi^\mu$$

As we are interested in minimal surfaces close to the original, we wish to impose the constraint that,

$$\mathrm{G}_j(\Sigma^\mu) = \mathrm{G}_j(\Sigma)$$

Let P_{B_j} be the unique holomorphic differential on Σ such that,

$$\int_{A_i} P_{B_j} = \delta_{i,j}$$

Since P_{B_j} and ψ are both type (1,0) forms, $P_{B_j} \wedge \psi = 0$. Since $\psi^\mu - \psi$ is regular with zero A_i-periods, and Proposition 3.4.2 and Corollary 3.4.5 give,

$$\begin{aligned}
\mathrm{Re} \int_\Sigma iP_{Bj} \wedge \psi^\mu &= \mathrm{Re} \int_\Sigma iP_{Bj} \wedge (\psi^\mu - \psi) && (3.15)\\
&= \mathrm{Re} \sum_i \left(\int_{A_i} iP_{Bj} \int_{B_i} (\psi^\mu - \psi) - \int_{A_i} (\psi^\mu - \psi) \int_{B_i} iP_{Bj} \right) && (3.16)\\
&= \mathrm{Re} \sum_i \left(\delta_{i,j} \int_{B_i} i(\psi^\mu - \psi) - 0 \int_{B_i} iP_{Bj} \right) && (3.17)\\
&= \mathrm{G}_j(\Sigma^\mu) - \mathrm{G}_j(\Sigma) && (3.18)
\end{aligned}$$

Applying the Proposition 3.3.2, we obtain,

$$\mathrm{G}_j(\Sigma^\mu) - \mathrm{G}_j(\Sigma) = \mathrm{Re} \int_\Sigma iP_{Bj} \wedge \psi^\mu = \mathrm{Re} \int_\Sigma (iP_{B_j}\psi)\mu + o(\|\mu\|^2)$$

Proposition 3.5.1. *The differential of* G_j *is represented by the holomorphic quadratic differential* $d_\mu|_\Sigma \mathrm{G}_j = i\psi P_{B_j}$.

3.6 Alignment of the zeroes of η with the zeroes and poles of g

The next constraints of Proposition 3.2.4 are that the zeroes of η^μ should coincide with the zeroes and poles of g^μ and have equal multiplicity. We now seek to find a function on Teichmüller space encoding this constraint and we seek its differential.

We choose a reference form R, holomorphic in a neighborhood of each puncture,

and for each pair z_j^+, we define $H_j : \mathrm{T}(\Sigma) \to \mathbf{R}$ and $I_j : \mathrm{T}(\Sigma) \to \mathbf{R}$ by,

$$\mathrm{H}_j(\Sigma^\mu) = \mathrm{Re}\frac{\eta}{R}(z_j^+)$$
$$\mathrm{I}_j(\Sigma^\mu) = -\mathrm{Im}\frac{\eta}{R}(z_j^+)$$

While these definitions depend upon the arbitrary choice of R, the set $H^{-1}(0)$ is a well defined subset of $\mathrm{T}(\Sigma)$.

Let Q_j be a one-form on Σ satisfying:

1. Q_j is regular except at z_j^+ where it has no residue and satisfies $Q_j = d\frac{\alpha}{R}$ for some locally meromorphic one form α with residue 1. Thus in coordinates, $Q_j = f(z)dz$ where f has a second order pole with no residue.

2. $\int_{A_i} Q_j = 0$

The existence of Q_j with such prescribed pole structure and A-periods is assured so long as the sum of the residues is zero. On a torus, we can just take R to be dz and Q_j to be some linear combination of dz and the Weierstrass $\wp$ function centered at z_j^0.

We again compute using Proposition 3.4.2 (bilinear relations),

$$\begin{aligned}\int_\Sigma Q_j \wedge \eta^\mu &= \int_\Sigma Q_j \wedge (\eta^\mu - \eta) \\ &= \sum_i \left(\int_{A_i} Q_j \int_{B_i} (\eta^\mu - \eta) - \int_{A_i} (\eta^\mu - \eta) \int_{B_i} Q_j \right) \\ &+ \lim_{\epsilon \to 0} \sum_p \int_{C_p^\epsilon D_p^\epsilon C_p^{\epsilon -1}} f(\eta^\mu - \eta)\end{aligned}$$

where f is a primitive of Q_j. Noting that neither Q_j nor $\eta^\mu - \eta$ has nonzero A_i-periods,

we see that

$$\sum_i \left(\int_{A_i} Q_j \int_{B_i} (\eta^\mu - \eta) - \int_{A_i} (\eta^\mu - \eta) \int_{B_i} Q_j \right) = 0$$

For $p = z_j^+$, we have, noting that $f = \frac{\alpha}{R}$,

$$\begin{aligned} \lim_{\epsilon \to 0} \int_{D_p^\epsilon} f(\eta^\mu - \eta) &= \lim_{\epsilon \to 0} \int_{D_p^\epsilon} \frac{\alpha}{R} (\eta^\mu - \eta) \\ &= \lim_{\epsilon \to 0} \int_{D_p^\epsilon} \alpha \frac{\eta^\mu - \eta}{R} \\ &= \frac{\eta^\mu - \eta}{R}(p) Res|_p \alpha \\ &= \frac{\eta^\mu}{R}(p) - \frac{\eta}{R}(p) \end{aligned}$$

Since Q_j has no residue at poles other than p,

$$\begin{aligned} \int_{C_p^\epsilon C_p^{\epsilon -1}} f(\eta^\mu - \eta) &= - \int_{C_p^\epsilon C_p^{\epsilon -1}} (\eta^\mu - \eta) \int_{D_i^\epsilon} Q_j \\ &= 0 \end{aligned}$$

At any other puncture p, since Q_j and $\eta^\mu - \eta$ are both regular there, Corollary 3.4.5 implies that,

$$\int_{C_p D_p C_p^{-1}} f(\eta^\mu - \eta) = 0$$

Collecting these results and applying the Rauch variational formula (Proposition

3.3.2), we obtain,

$$\begin{aligned}(\mathrm{H}_j - i\mathrm{I}_j)(\Sigma^\mu) - (\mathrm{H}_j - i\mathrm{I}_j)(\Sigma) &= \frac{\eta^\mu}{R}(p) - \frac{\eta}{R}(p) \\ &= \int_\Sigma Q_j \wedge \eta^\mu \\ &= \int_\Sigma (Q_j\eta)\mu + o(\|\mu\|^2)\end{aligned}$$

Proposition 3.6.1. *The complex differential of* $\mathrm{H}_j - i\mathrm{I}_j$ *is represented by the quadratic differential* $d(\mathrm{H}_j - i\mathrm{I}_j) = Q_j\eta$. *The real differentials of* H_j *and* I_j *are* $d\mathrm{H}_j = Q_j\eta$ *and* $d\mathrm{I}_j = iQ_j\eta$, *respectively.*

3.7 An expression for g^μ

In what follows, we will have need for formulae relating $g^\mu(p)$ to $g(p)$ at punctures p. From our definition of g^μ in terms of ψ^μ, we have,

$$g^\mu(p) = e^{\int_{e_1^-}^p \psi^\mu} \tag{3.19}$$

$$= \left(e^{\int_{e_1^-}^p \psi}\right) e^{\int_{e_1^-}^p \psi^\mu - \psi} \tag{3.20}$$

$$= g(p)e^{\int_{e_1^-}^p \psi^\mu - \psi} \tag{3.21}$$

For each puncture $q \neq e_1^-$, we let P_q be the meromorphic one form on Σ satisfying the following:

1. P_q is regular except at e_1^- and q, where it has first order poles with residues $\frac{1}{2\pi i}$ and $-\frac{1}{2\pi i}$, respectively.

2. $\int_{A_i} P_q = 0$

We apply bilinear relations (Proposition 3.4.2), and find (with $P_q = df$),

$$\begin{aligned}\int_\Sigma P_q \wedge \psi^\mu &= \int_\Sigma P_q \wedge (\psi^\mu - \psi) \\ &= \sum_i \left(\int_{A_i} P_q \int_{B_i} (\psi^\mu - \psi) - \int_{A_i} (\psi^\mu - \psi) \int_{B_i} P_q \right) \\ &\quad + \lim_{\epsilon \to 0} \sum_p \int_{C_p^\epsilon D_p^\epsilon C_p^{\epsilon -1}} f(\psi^\mu - \psi)\end{aligned}$$

Since $\psi^\mu - \psi$ has zero A_i-periods, we have,

$$\begin{aligned}&\sum_i \left(\int_{A_i} P_q \int_{B_i} (\psi^\mu - \psi) - \int_{A_i} (\psi^\mu - \psi) \int_{B_i} P_q \right) \\ &\quad = \sum_i \left(0 \int_{B_i} (\psi^\mu - \psi) - 0 \int_{B_i} P_q \right) \\ &\quad = 0\end{aligned}$$

Since $\psi^\mu - \psi$ is everywhere regular and P_q has a simple poles at e_1^- and q, we find by Corollary 3.4.4 that,

$$\begin{aligned}\lim_{\epsilon \to 0} \int_{C^\epsilon_{e_1^-} D^\epsilon_{e_1^-} C^{\epsilon -1}_{e_1^-}} f(\psi^\mu - \psi) &= -2\pi i \mathrm{Res}|_{e_1^-} P_q \int_{C_{e_1^-}} (\psi^\mu - \psi) \\ \lim_{\epsilon \to 0} \int_{C_q^\epsilon D_q^\epsilon C_q^{\epsilon -1}} f(\psi^\mu - \psi) &= -2\pi i \mathrm{Res}|_q P_q \int_{C_q} (\psi^\mu - \psi)\end{aligned}$$

We combine both of these terms into:

$$\lim_{\epsilon \to 0} \left(\int_{C^\epsilon_{e_1^-} D^\epsilon_{e_1^-} C^{\epsilon -1}_{e_1^-}} f(\psi^\mu - \psi) + \int_{C_q^\epsilon D_q^\epsilon C_q^{\epsilon -1}} f(\psi^\mu - \psi) \right) = \int_{e_1^-}^q (\psi^\mu - \psi)$$

Since P_q is regular elsewhere, the remaining terms are zero. Collecting terms, we find,

$$\int_{e_1^-}^{q} (\psi^\mu - \psi) = \int_\Sigma P_q \wedge \psi^\mu$$

Substituting this back into (3.19) yields,

$$g^\mu(q) = g(q) e^{\int_\Sigma P_q \wedge \psi^\mu}$$

By our Rauch variational formula (Proposition 3.3.2), we find that,

$$\begin{aligned} g^\mu(q) &= g(q) e^{\int_\Sigma P_q \wedge (\psi\mu + o(\|\mu\|^2))} \\ &= g(p) \left(1 + \int_\Sigma P_q \wedge \psi\mu\right) + o(\|\mu\|^2) \end{aligned}$$

This is enough to prove,

Proposition 3.7.1. *The holomorphic quadratic differential representing the differential of $g^\mu(q)$ is:*

$$d_\mu|_\Sigma g^\mu(q) = g(q) P_q \psi$$

Finally, for convenience of expression, we take $P_{e_1^-} = 0$. Since $g^\mu(e_1^-) = 1$ is fixed, the above proposition holds if we take q to be any puncture, including e_1^-.

3.8 Periods of ϕ_1 and ϕ_2

In the previous sections, we have defined a number of functions on Teichmüller space. We now seek to impose constraints that will enforce the remaining conditions needed for a Scherk-type minimal surface. That is, our data must satisfy the so-called period

problem. For this purpose, it useful to consider a sublocus of Teichmüller space on which the previous constraints are all satsified. Namely, we let $\mathrm{T}^{\alpha}(\Sigma)$ be the subset of $\mathrm{T}^{\sigma}(\Sigma)$ such that:

- $\mathrm{G}_j(\Sigma^{\mu}) = 0$, implying that g^{μ} exists.
- $\mathrm{H}_k(\Sigma^{\mu}) = 0$ and $\mathrm{I}_k(\Sigma^{\mu}) = 0$, implying that the zeroes of η^{μ} coincide with the poles of ψ^{μ} (which are the poles and zeroes of g^{μ}).

We then let J_j and K_j be the functions $\mathrm{T}^{\alpha}(\Sigma) \rightarrow \mathbf{R}$ given by,

$$J_j(\Sigma^{\mu}) = \mathrm{Re} \int_{B_j} \frac{1}{2}(\frac{1}{g^{\mu}} - g^{\mu})\eta^{\mu}$$

$$K_j(\Sigma^{\mu}) = \mathrm{Re} \int_{B_j} \frac{i}{2}(\frac{1}{g^{\mu}} + g^{\mu})\eta^{\mu}$$

Let S_j be the one-forms on Σ satisfying:

1. S_j is regular except at z_1^+ where it has a second order pole with no residue.
2. $\int_{A_i} S_j = \delta_{i,j}$
3. $\int_{B_j} S_j = 0$

Then we let $T_j = -\overline{\sigma^* S_j}$, and verify that:

1. T_j is regular except at z_1^- where it has a second order pole with no residue.
2. $\int_{A_i} T_j = \int_{A_i} -\overline{\sigma^* S_j} = \overline{\int_{\sigma(A_i)} -S_j} = \overline{\int_{A_i} S_j} = \delta_{i,j}$
3. $\int_{B_j} T_j = 0$

Note: Unlike the forms Q_j in the previous section, it is not clear that the S_j and T_j exist for every Riemann surface. However, we again have a simple construction of such forms on tori, taking linear combinations of Weierstrass $\wp dz$ and dz, so long as $\int_B \wp dz \neq 0$.

We compute using bilinear relations (Proposition 3.4.2), where $df = S_j$,

$$\begin{aligned}\int_\Sigma S_j \wedge g^\mu \eta^\mu &= \int_\Sigma S_j \wedge (g^\mu \eta^\mu - g\eta) \\ &= \sum_i \left(\int_{A_i} S_j \int_{B_i} (g^\mu \eta^\mu - g\eta) - \int_{A_i} (g^\mu \eta^\mu - g\eta) \int_{B_i} S_j \right) \\ &\quad + \lim_{\epsilon \to 0} \sum_p \int_{C_p^\epsilon D_p^\epsilon C_p^{\epsilon -1}} f(g^\mu \eta^\mu - g\eta)\end{aligned}$$

Similarly, we compute, where $df' = T_j$,

$$\begin{aligned}\int_\Sigma T_j \wedge \frac{1}{g^\mu}\eta^\mu &= \int_\Sigma T_j \wedge (\frac{1}{g^\mu}\eta^\mu - \frac{1}{g}\eta) \\ &= \sum_i \left(\int_{A_i} T_j \int_{B_i} (\frac{1}{g^\mu}\eta^\mu - \frac{1}{g}\eta) - \int_{A_i} (\frac{1}{g^\mu}\eta^\mu - \frac{1}{g}\eta) \int_{B_i} T_j \right) \\ &\quad + \lim_{\epsilon \to 0} \sum_p \int_{C_p^\epsilon D_p^\epsilon C_p^{\epsilon -1}} f'(\frac{1}{g^\mu}\eta^\mu - g\eta)\end{aligned}$$

We simplify the first term of each,

$$\begin{aligned}&\sum_i \left(\int_{A_i} S_j \int_{B_i} (g^\mu \eta^\mu - g\eta) - \int_{A_i} (g^\mu \eta^\mu - g\eta) \int_{B_i} S_j \right) \\ &\quad = \sum_i \left(\delta_{i,j} \int_{B_i} (g^\mu \eta^\mu - g\eta) - 0 \int_{A_i} (g^\mu \eta^\mu - g\eta) \right) \\ &\quad = \int_{B_j} (g^\mu \eta^\mu - g\eta)\end{aligned}$$

$$\sum_i \left(\int_{A_i} T_j \int_{B_i} (\frac{1}{g^\mu}\eta^\mu - \frac{1}{g}\eta) - \int_{A_i} (\frac{1}{g^\mu}\eta^\mu - \frac{1}{g}\eta) \int_{B_i} T_j \right)$$
$$= \sum_i \left(\delta_{i,j} \int_{B_i} (\frac{1}{g^\mu}\eta^\mu - \frac{1}{g}\eta) - 0 \int_{A_i} (\frac{1}{g^\mu}\eta^\mu - \frac{1}{g}\eta) \right)$$
$$= \int_{B_j} (\frac{1}{g^\mu}\eta^\mu - \frac{1}{g}\eta)$$

For $p = e_i^+$ or $p = e_i^-$, we have,

$$\lim_{\epsilon \to 0} \int_{C_p^\epsilon D_p^\epsilon C_p^{\epsilon -1}} f(g^\mu \eta^\mu - g\eta) = \lim_{\epsilon \to 0} \left(\int_{C_p^\epsilon D_p^\epsilon C_p^{\epsilon -1}} f g^\mu \eta^\mu - \int_{C_p^\epsilon D_p^\epsilon C_p^{\epsilon -1}} fg\eta \right)$$
$$= 2\pi i \left(\int_{C_p} S_j \right) (\mathrm{Res}|_p(g^\mu \eta^\mu) - \mathrm{Res}|_p(g\eta))$$
$$= 2\pi i \left(\int_{C_p} S_j \right) (g^\mu(p)\mathrm{Res}|_p \eta^\mu - g(p)\mathrm{Res}|_p \eta)$$
$$= 2\pi i \mathrm{Res}|_p \eta \left(\int_{C_p} S_j \right) (g^\mu(p) - g(p))$$

$$\lim_{\epsilon \to 0} \int_{C_p^\epsilon D_p^\epsilon C_p^{\epsilon -1}} f'(\frac{1}{g^\mu}\eta^\mu - \frac{1}{g}\eta) = \lim_{\epsilon \to 0} \left(\int_{C_p^\epsilon D_p^\epsilon C_p^{\epsilon -1}} f' \frac{1}{g^\mu}\eta^\mu - \int_{C_p^\epsilon D_p^\epsilon C_p^{\epsilon -1}} f' \frac{1}{g}\eta \right)$$
$$= 2\pi i \left(\int_{C_p} T_j \right) (\mathrm{Res}|_p(\frac{1}{g^\mu}\eta^\mu) - \mathrm{Res}|_p(\frac{1}{g}\eta))$$
$$= 2\pi i \left(\int_{C_p} T_j \right) (\frac{1}{g^\mu(p)}\mathrm{Res}|_p \eta^\mu - \frac{1}{g(p)}\mathrm{Res}|_p \eta)$$
$$= 2\pi i \mathrm{Res}|_p \eta \left(\int_{C_p} T_j \right) (\frac{1}{g^\mu(p)} - \frac{1}{g(p)})$$

For $p = z_i^+$, we have,

$$\lim_{\epsilon \to 0} \int_{C_p^\epsilon C_p^{\epsilon -1}} f(g^\mu \eta^\mu - g\eta) = 0$$

since S has no residue at p. Also,

$$\lim_{\epsilon \to 0} \int_{D_p^\epsilon} f(g^\mu \eta^\mu - g\eta) = 0$$

since f has a simple pole (being a primitive for S which has a second order pole with no residue) and both g^μ and g have zeroes at p. We have,

$$\lim_{\epsilon \to 0} \int_{C_p^\epsilon C_p^{\epsilon^{-1}}} f'(\frac{1}{g^\mu}\eta^\mu - \frac{1}{g}\eta) = 0$$

since T is regular at p, and on the locus with $\eta(p) = 0$, have,

$$\lim_{\epsilon \to 0} \int_{D_p^\epsilon} f'(\frac{1}{g^\mu}\eta^\mu - \frac{1}{g}\eta) = f'(p)(\mathrm{Res}|_p \frac{1}{g^\mu}\eta^\mu - \mathrm{Res}|_p \frac{1}{g}\eta) = 0$$

Similarly, for $p = z_i^-$, we have,

$$\lim_{\epsilon \to 0} \int_{C_p^\epsilon C_p^{\epsilon^{-1}}} f'(\frac{1}{g^\mu}\eta^\mu - \frac{1}{g}\eta) = 0$$

since T has no residue at p, and,

$$\lim_{\epsilon \to 0} \int_{D_p^\epsilon} f'(\frac{1}{g^\mu}\eta^\mu - \frac{1}{g}\eta) = 0$$

since f' has a simple pole (being a primitive for T which has a second order pole with no residue) and both g^μ and g have zeroes at p. We have,

$$\lim_{\epsilon \to 0} \int_{C_p^\epsilon C_p^{\epsilon^{-1}}} f(g^\mu \eta^\mu - g\eta) = 0$$

since S is regular at p, and on the locus with $\eta(p) = 0$, have,

$$\lim_{\epsilon \to 0} \int_{D_p^\epsilon} f(g^\mu \eta^\mu - g\eta) = f(p)(\mathrm{Res}|_p g^\mu \eta^\mu - \mathrm{Res}|_p g\eta) = 0$$

Both S and $g^\mu \eta^\mu - g\eta$ are regular at $p = z_i^-$, so that term vanishes. Both T and $\frac{1}{g^\mu}\eta^\mu - \frac{1}{g}\eta$ are regular at $p = z_i^+$, so that term vanishes. Collecting contributions we find that:

$$\begin{aligned}
&\int_{B_j} (g^\mu \eta^\mu - g\eta) \\
&= \int_\Sigma S_j \wedge g^\mu \eta^\mu - 2\pi i \sum_i \Big[Res|_{e_i^+} \eta \left(\int_{C_{e_i^+}} S_j \right) \left(g^\mu(e_i^+) - g(e_i^+) \right) \\
&\qquad + Res|_{e_i^-} \eta \left(\int_{C_{e_i^-}} S_j \right) \left(g^\mu(e_i^-) - g(e_i^-) \right) \Big] \\
&= \int_\Sigma S_j \wedge g^\mu \eta^\mu - \sum_i \Big[\left(\int_{C_{e_i^+}} S_j \right) \left(g^\mu(e_i^+) - g(e_i^+) \right) \\
&\qquad - \left(\int_{C_{e_i^-}} S_j \right) \left(g^\mu(e_i^-) - g(e_i^-) \right) \Big]
\end{aligned}$$

$$\int_{B_j} (\frac{1}{g^\mu}\eta^\mu - \frac{1}{g}\eta)$$

$$= \int_\Sigma T_j \wedge \frac{1}{g^\mu}\eta^\mu - 2\pi i \sum_i \Big[Res|_{e_i^+}\eta \left(\int_{C_{e_i^+}} T_j\right)\left(\frac{1}{g^\mu(e_i^+)} - \frac{1}{g(e_i^+)}\right)$$

$$+ Res|_{e_i^-}\eta \left(\int_{C_{e_i^-}} T_j\right)\left(\frac{1}{g^\mu(e_i^-)} - \frac{1}{g(e_i^-)}\right)\Big]$$

$$= \int_\Sigma T_j \wedge \frac{1}{g^\mu}\eta^\mu - \sum_i \Big[\left(\int_{C_{e_i^+}} T_j\right)\left(\frac{1}{g^\mu(e_i^+)} - \frac{1}{g(e_i^+)}\right) -$$

$$\left(\int_{C_{e_i^-}} T_j\right)\left(\frac{1}{g^\mu(e_i^-)} - \frac{1}{g(e_i^-)}\right)\Big]$$

Before continuing, we note that, as $C_{e_i^+}$ is homologous to $\sigma(C_{e_i^+})$,

$$\int_{C_{e_i^+}} T_j = \int_{C_{e_i^+}} -\overline{\sigma^* S_j} = -\overline{\int_{\sigma(C_{e_i^+})} \sigma^* S_j} = -\overline{\int_{C_{e_i^+}} S_j}$$

and that, at any fixed point of σ,

$$\frac{1}{g^\mu} = \overline{\sigma_* g^\mu} = \overline{g^\mu}$$

We now compute,

$$
\begin{aligned}
&J_j(\Sigma^\mu) - J_j(\Sigma) \\
&\quad = \mathrm{Re}\frac{1}{2}\Big[\int_{B_i}(\frac{1}{g^\mu}\eta^\mu - \frac{1}{g}\eta) - \int_{B_i}((g^\mu)\eta^\mu - g\eta)\Big] \\
&\quad = \mathrm{Re}\frac{1}{2}\Big[\int_\Sigma T_j \wedge \frac{1}{g^\mu}\eta^\mu - \sum_i \Big[\left(\int_{C_{e_i^+}} T_j\right)\left(\frac{1}{g^\mu(e_i^+)} - \frac{1}{g(e_i^+)}\right) \\
&\qquad\qquad - \left(\int_{C_{e_i^-}} T_j\right)\left(\frac{1}{g^\mu(e_i^-)} - \frac{1}{g(e_i^-)}\right)\Big] \\
&\qquad - \int_\Sigma S_j \wedge g^\mu\eta^\mu + \sum_i \Big[\left(\int_{C_{e_i^+}} S_j\right)\left(g^\mu(e_i^+) - g(e_i^+)\right) \\
&\qquad\qquad - \left(\int_{C_{e_i^-}} S_j\right)\left(g^\mu(e_i^-) - g(e_i^-)\right)\Big]\Big] \\
&\quad = \mathrm{Re}\frac{1}{2}\int_\Sigma T_j \wedge \frac{1}{g^\mu}\eta^\mu - S_j \wedge g^\mu\eta^\mu \\
&\quad - \mathrm{Re}\frac{1}{2}\sum_i \Big[\left(\int_{C_{e_i^+}} T_j\right)\left(\frac{1}{g^\mu(e_i^+)} - \frac{1}{g(e_i^+)}\right) - \left(\int_{C_{e_i^-}} T_j\right)\left(\frac{1}{g^\mu(e_i^-)} - \frac{1}{g(e_i^-)}\right) \\
&\qquad - \left(\int_{C_{e_i^+}} S_j\right)\left(g^\mu(e_i^+) - g(e_i^+)\right) + \left(\int_{C_{e_i^-}} S_j\right)\left(g^\mu(e_i^-) - g(e_i^-)\right)\Big] \\
&\quad = \mathrm{Re}\frac{1}{2}\int_\Sigma T_j \wedge \frac{1}{g^\mu}\eta^\mu - S_j \wedge g^\mu\eta^\mu \\
&\quad - \mathrm{Re}\frac{1}{2}\sum_i \Big[-\left(\int_{C_{e_i^+}} S_j\right)\left(g^\mu(e_i^+) - g(e_i^+)\right) + \left(\int_{C_{e_i^-}} S_j\right)\left(g^\mu(e_i^-) - g(e_i^-)\right) \\
&\qquad - \left(\int_{C_{e_i^+}} S_j\right)\left(g^\mu(e_i^+) - g(e_i^+)\right) + \left(\int_{C_{e_i^-}} S_j\right)\left(g^\mu(e_i^-) - g(e_i^-)\right)\Big] \\
&\quad = \mathrm{Re}\frac{1}{2}\int_\Sigma T_j \wedge \frac{1}{g^\mu}\eta^\mu - S_j \wedge g^\mu\eta^\mu \\
&\quad + \mathrm{Re}\sum_i \Big[\left(\int_{C_{e_i^+}} S_j\right)\left(g^\mu(e_i^+) - g(e_i^+)\right) - \left(\int_{C_{e_i^-}} S_j\right)\left(g^\mu(e_i^-) - g(e_i^-)\right)\Big]
\end{aligned}
$$

Similarly, we have,

$$
\begin{aligned}
&K_j(\Sigma^\mu) - K_j(\Sigma) \\
&\quad = \mathrm{Re}\frac{i}{2}\Big[\int_{B_i}(\frac{1}{g^\mu}\eta^\mu + \frac{1}{g}\eta) - \int_{B_i}((g^\mu)\eta^\mu - g\eta)\Big] \\
&\quad = \mathrm{Re}\frac{i}{2}\Big[\int_\Sigma T_j \wedge \frac{1}{g^\mu}\eta^\mu - \sum_i \Big[\left(\int_{C_{e_i^+}} T_j\right)\left(\frac{1}{g^\mu(e_i^+)} - \frac{1}{g(e_i^+)}\right) \\
&\qquad\qquad - \left(\int_{C_{e_i^-}} T_j\right)\left(\frac{1}{g^\mu(e_i^-)} - \frac{1}{g(e_i^-)}\right)\Big] \\
&\qquad + \int_\Sigma S_j \wedge g^\mu\eta^\mu + \sum_i \Big[\left(\int_{C_{e_i^+}} S_j\right)\left(g^\mu(e_i^+) - g(e_i^+)\right) \\
&\qquad\qquad - \left(\int_{C_{e_i^-}} S_j\right)\left(g^\mu(e_i^-) - g(e_i^-)\right)\Big]\Big] \\
&\quad = \mathrm{Re}\frac{i}{2}\int_\Sigma T_j \wedge \frac{1}{g^\mu}\eta^\mu + S_j \wedge g^\mu\eta^\mu \\
&\quad - \mathrm{Re}\frac{i}{2}\sum_i \Big[\left(\int_{C_{e_i^+}} T_j\right)\left(\frac{1}{g^\mu(e_i^+)} - \frac{1}{g(e_i^+)}\right) - \left(\int_{C_{e_i^-}} T_j\right)\left(\frac{1}{g^\mu(e_i^-)} - \frac{1}{g(e_i^-)}\right) \\
&\qquad + \left(\int_{C_{e_i^+}} S_j\right)\left(g^\mu(e_i^+) - g(e_i^+)\right) - \left(\int_{C_{e_i^-}} S_j\right)\left(g^\mu(e_i^-) - g(e_i^-)\right)\Big] \\
&\quad = \mathrm{Re}\frac{i}{2}\int_\Sigma T_j \wedge \frac{1}{g^\mu}\eta^\mu + S_j \wedge g^\mu\eta^\mu \\
&\quad - \mathrm{Re}\frac{i}{2}\sum_i \Big[-\left(\int_{C_{e_i^+}} S_j\right)\left(g^\mu(e_i^+) - g(e_i^+)\right) + \left(\int_{C_{e_i^-}} S_j\right)\left(g^\mu(e_i^-) - g(e_i^-)\right) \\
&\qquad + \left(\int_{C_{e_i^+}} S_j\right)\left(g^\mu(e_i^+) - g(e_i^+)\right) - \left(\int_{C_{e_i^-}} S_j\right)\left(g^\mu(e_i^-) - g(e_i^-)\right)\Big] \\
&\quad = \mathrm{Re}\frac{i}{2}\int_\Sigma T_j \wedge \frac{1}{g^\mu}\eta^\mu + S_j \wedge g^\mu\eta^\mu \\
&\quad - \mathrm{Re}i\sum_i \Big[\left(\int_{C_{e_i^+}} S_j\right)\left(g^\mu(e_i^+) - g(e_i^+)\right) + \left(\int_{C_{e_i^-}} S_j\right)\left(g^\mu(e_i^-) - g(e_i^-)\right)\Big]
\end{aligned}
$$

Applying our Rauch variational formula (Proposition 3.3.2) and our formula for the differential of g^μ (Proposition 3.7.1) we obtain,

Proposition 3.8.1. *The holomorphic quadratic differentials representing the differentials of J_j and K_j on the sublocus $\mathrm{T}^\alpha(\Sigma)$ are given by:*

$$d_\mu|_\Sigma J_j = \frac{1}{2}(T_j\frac{1}{g} - S_j g)\eta + \sum_i \Big[\left(\int_{C_{e_i^+}} S_j\right) g(e_i^+)P_{e_i^+} - \left(\int_{C_{e_i^-}} S_j\right) g(e_i^-)P_{e_i^-}\Big]\psi$$

$$d_\mu|_\Sigma K_j = \frac{i}{2}(T_j\frac{1}{g} + S_j g)\eta - i\sum_i \Big[\left(\int_{C_{e_i^+}} S_j\right) g(e_i^+)P_{e_i^+} + \left(\int_{C_{e_i^-}} S_j\right) g(e_i^-)P_{e_i^-}\Big]\psi$$

3.9 From independence to a surface family

Proposition 3.9.1. *There exists $T^\alpha(\Sigma)$, an $2n_g + 2n_c - 3$ dimensional submanifold of $\mathrm{T}^\sigma(\Sigma)$ in a neighborhood of Σ, on which*

- $G_j(\Sigma^\mu) = 0$, *implying the existence of g^μ*
- $H_j(\Sigma^\mu) = 0$, *implying $Re\,\eta^\mu(z_j^-) = Re\,\eta^\mu(z_j^+) = 0$*
- $I_j(\Sigma^\mu) = 0$, *implying $Re\,\eta^\mu(z_j^-) = Re\,\eta^\mu(z_j^+) = 0$*

if the following quadratic differentials are linearly independent:

- $i\psi P_{B_j}$, $1 \le j \le n_g$
- $Q_j^+\eta$, $1 \le j \le n_z$
- $iQ_j^-\eta$, $1 \le j \le n_z$

Proof. We consider the map,

$$f_1 : \mathrm{T}^\sigma(\Sigma) \to R^{n_g+n_z+n_z}$$
$$f_1(\Sigma^\mu) = (G(\Sigma^\mu), H(\Sigma^\mu), I(\Sigma^\mu))$$

The differential of G_j was shown to be $iP_{B_j}\psi$. Since $iP_{B_j}\psi$ has no poles and $\mathrm{Im}\int_{A_i} iP_{B_j} = 0$, we have that $P^-_{B_j} = 0$. Since ψ has $\mathrm{Re}\int_{A_i}\psi = 0$, $\mathrm{Im}Res_p\psi = 0$ for $p = e_i^-$ and $p = e_i^+$, and no other poles, we have that $\psi^+ = 0$. It follows that $(d_\mu|_\Sigma \mathrm{G}_j)^+ = (iP_{B_j}\psi)^+ = 0$. Thus the differential to G at Σ is antisymmetric, and its projection to the cotangent space of $\mathrm{T}^\sigma(\Sigma)$ is trivial.

The differential of H_j was shown to be $Q_j\eta$. The projection of this differential to the cotangent space of $\mathrm{T}^\sigma(\Sigma)$ is $(Q_j\eta)^-$. Since η has been shown to be antisymmetric, it follows that

$$(Q_j\eta)^- = ((Q_j^+ + Q_j^-)\eta)^- = Q_j^+\eta$$

The differential of I_j was shown to be $iQ_j\eta$. The projection of this differential to the cotangent space of $\mathrm{T}^\sigma(\Sigma)$ is $(iQ_j\eta)^-$. Since η has been shown to be antisymmetric, it follows that $i\eta$ is symmetric, and,

$$(iQ_j\eta)^- = ((Q_j^+ + Q_j^-)i\eta)^- = iQ_j^-\eta$$

Thus, each of the differentials in our list is anti-symmetric, thus if they are independent in $\mathrm{T}(\Sigma)$ (equivalently, independent as holomorphic quadratic differentials) then they project to independent covectors in $\mathrm{T}^\alpha(\Sigma)$. So assume the differentials are independent. It then follows that the map $df_1 : T_*(\mathrm{T}^\sigma(\Sigma)) \to T_*(R^{n_g+n_z+n_z})$ is surjective. By assumption, $f_1(0) = 0$, and so by the local submersion theorem

there exist coordinates $x_1, ...x_{(3n_g+2n_z+2n_e-3)}$ on $\mathrm{T}^\sigma(\Sigma)$, and coordinates $y_1, ..., y_{(n_g-2n_z)}$ on $R^{n_g-2n_z}$, such that, in coordinates, $f_1(x_1, ..., x_{(3n_g+2n_z+2n_e-3)}) = (x_1, ..., x_{n_g-2n_z})$. It follows that $f_1^{-1}(0, ..., 0)$ is locally a manifold of dimension $(3n_g + 2n_z + 2n_e - 3) - n_g - 2n_z = 2n_g + 2n_e - 3$, call it $\mathrm{T}^\alpha(\Sigma)$, parameterized by the coordinates $x_{(n_g-2n_z+1)}, \cdots, x_{(3n_g+2n_z+2n_e-3)}$. □

We then apply the local submersion theorem again.

Proposition 3.9.2. *There exists an $2n_e - 3$ dimensional submanifold in a neighborhood of Σ satisfying the following constraints:*

- $G_j(\Sigma^\mu) = 0$, *implying the existence of* g^μ
- $H_j(\Sigma^\mu) = 0$, *implying* $Re\,\eta^\mu(z_j^-) = Re\,\eta^\mu(z_j^+) = 0$
- $I_j(\Sigma^\mu) = 0$, *implying* $Re\,\eta^\mu(z_j^-) = Re\,\eta^\mu(z_j^+) = 0$
- $J_j(\Sigma^\mu) = 0$, *implying that the period problem is solved for* ϕ_1 *around* B_j
- $K_j(\Sigma^\mu) = 0$, *implying that the period problem is solved for* ϕ_2 *around* B_j

if the following set of quadratic differentials is linearly independent:

- $i\psi P_{B_j}$, $1 \le j \le n_g$
- $Q_j^+\eta$, $1 \le j \le n_z$
- $iQ_j^-\eta$, $1 \le j \le n_z$
- $\frac{1}{2}(T_j\frac{1}{g} - S_j g)\eta + \sum_i \left[c_1(e_i^+)P_{e_i^+} - c_1(e_i^-)P_{e_i^-}\right]\psi_j$, $1 \le j \le n_g$
- $\frac{i}{2}(T_j\frac{1}{g} + S_j g)\eta - i\sum_i \left[c_2(e_i^+)P_{e_i^+} + c_2(e_i^-)P_{e_i^-}\right]\psi_j$, $1 \le j \le n_g$

where

$$c_1(p) = Re\left(\int_{C_p} S_j\right) g(p)$$
$$c_2(p) = Im\left(\int_{C_p} S_j\right) g(p)$$

Proof. We consider the map,

$$f_2 : T^\alpha(\Sigma) \to R^{n_g+n_z+n_z}$$
$$f_2(\Sigma^\mu) = (J(\Sigma^\mu), K(\Sigma^\mu))$$

We've computed the differential to have components of the form:

$$\Phi_1 = d_\mu|_\Sigma J_j = \frac{1}{2}(T_j\frac{1}{g} - S_j g)\eta + \sum_i \Big[\left(\int_{C_{e_i^+}} S_j\right) g(e_i^+)P_{e_i^+} - \left(\int_{C_{e_i^-}} S_j\right) g(e_i^-)P_{e_i^-}\Big]\psi_j$$
$$\Phi_2 = d_\mu|_\Sigma K_j = \frac{i}{2}(T_j\frac{1}{g} + S_j g)\eta - i\sum_i \Big[\left(\int_{C_{e_i^+}} S_j\right) g(e_i^+)P_{e_i^+} + \left(\int_{C_{e_i^-}} S_j\right) g(e_i^-)P_{e_i^-}\Big]\psi_j$$

We recall that

$$\overline{\sigma^* T_j} = -S_j$$
$$\overline{\sigma^* g} = \frac{1}{g}$$

and compute that,

$$\overline{\sigma^*(T_j\frac{1}{g} - S_j g)} = T_j\frac{1}{g} - S_j g$$
$$\overline{\sigma^* i(T_j\frac{1}{g} + S_j g)} = i(T_j\frac{1}{g} + S_j g)$$

Thus these factors are symmetric. Since the η factors are antisymmetric, we have that $\frac{1}{2}(T_j\frac{1}{g} - S_j g)\eta$ and $\frac{i}{2}(T_j\frac{1}{g} + S_j g)\eta$ are antisymmetric.

For the summation term, we note that the A_i-periods of $P_{e_i^\pm}$ are 0, and the poles of $P_{e_i^-}$, $i \neq 1$, or $P_{e_i^+}$ lie at $e_i^\pm$ and are real. Thus $P_{e_i^\pm}$ is symmetric. Since the ψ_j were shown antisymmetric in the proof of the preceeding proposition, we have that $P_{e_i^\pm}\psi_j$ is antisymmetric, and thus the coefficient of it is antisymmetric component will be real.

We have shown that Φ_1 and Φ_2 are anti-symmetric. Now assume they are independent. It follows that, for any $v \in T_*(R^{2n_e-3})$, there exists a vector $\nu \in T_*(\mathrm{T}^\sigma(\Sigma))$ such that $df_2\nu = v$. If, furthermore, Φ_1 and Φ_2 are independent from the other differentials on the list, we can project such a ν to $\nu^\alpha \in T_*(\mathrm{T}^\sigma(\Sigma))$, such that $df_2\nu^\alpha = v$. By assumption, $f_2(0) = 0$, and so by the local submersion theorem, there exists coordinates $x_1, ...x_{(2n_g+2n_e-3)}$ on $\mathrm{T}^\sigma(\Sigma)$, and coordinates $y_1, ..., y_{2n_g}$ on $T_*(R^{2n_g})$, such that $f_2(x_1, ..., x_{(2n_g+2n_e-3)}) = (x_1, ..., x_{2n_g})$. It follows that $f_2^{-1}(0, ..., 0)$ is locally a manifold of dimension $(2n_g + 2n_e - 3) - 2n_g = 2n_e - 3$, parameterized by the coordinates $x_{(2n_g+1)}, \cdots, x_{(2n_g+2n_e-3)}$. □

Ideally, we would like to establish conditions for the independence of these quadratic differentials analytically, but the implicit representation we have for the domain of Karcher's surface (as guaranteed by the intermediate value theorem), makes this difficult. Barring that, however, we have developed numerical methods to numerically

prove their independence at particular surfaces. We now develop some machinery for computing on tori and apply it to the question of the independence of the above quadratic differentials on Karcher's surface.

Chapter 4

Numerical Methods

4.1 Implementation Details

The numerical portion of this work makes heavy use of interval arithmetic to establish conservative bounds on values that cannot be computed exactly. Interval arithmetic has been studied extensively (see [Moore79] or [Neumaier90]). While there is a rich body of knowledge surrounding intervals, our work requires only their elementary properties, most notably:

Theorem 4.1.1. *(Moore's Fundamental Theorem) Suppose we have an expression defining a function f as a combination of elementary functions. If each elementary function can be approximated conservatively on intervals (meaning that given intervals for the inputs to the elementary function, an interval can be found which contains their image), then f may be also approximated convervatively on intervals by chaining these elementary approximations together.*

Proof. The proof is just an induction on the size of the expression. □

We use the Boost C++ Interval library (see [Boost]) for interval arithmetic. In Figure 4.1, we define three number types that will be used in our computations. The first definition creates a type named `FP` as an alias for the standard C++ floating-point type `long double`. We will systematically use `FP` whenever we need a floating-point type.

```
typedef long double FP;

typedef interval<FP,
        policies<save_state<rounded_transc_opp<FP> >,
            checking_no_nan<FP, checking_no_empty<FP> > > > R;

typedef complex<R> C;
C exp(C z)
{
    return C(cos(z.imag()), sin(z.imag())) * exp(z.real());
}

C errorTerm(C z)
{
    // nub is a norm upper bound for z
    FP nub = max(fabs(z.real().lower()), fabs(z.real().upper()))
           + max(fabs(z.imag().lower()), fabs(z.imag().upper()));
    return C(R(-nub, nub), R(-nub, nub));
}

C pow(C base, int exp)
{
    C result(1);
    for(int i=0; i<exp; ++i)
        result *= base;
    return result;
}
```

Figure 4.1 : Definitions of basic numerical types

The second definition creates a type `R` as an alias for a Boost interval whose endpoints are of type `FP`. The policy `save_state<rounded_transc_opp<FP> >` specifies a version of operations be used which attempts to avoid changing the floating point rounding direction (lower bounds need to be rounded down, upper bounds up) by apppropriate use of negation. This is just an optimization and shall not concern us further. The policy `checking_no_nan` specifies that `NaN` (a special floating-point value meaning "not a number") should not ever arise during a computation. The policy `checking_no_empty` specifies that no interval computed should ever be empty. If either of these conditions arises, an `std::runtime_error` exception is thrown and the program exits. A more detailed description of the meanings of these policies can be found in the Boost documentation.

The final definition creates a complex type `C` as an alias for the C++ standard library class `std::complex` with intervals of type `R` for its real and imaginary parts.

With these definitions, we may use the operator `+`, `-`, `*`, and `/` to perform complex interval addition, subtraction, multiplication and division on values of type `C`. There are a few operations we will need to build ourselves, several of which are shown in Figure 4.1. The function `exp(C z)` computes a complex exponential function using real interval functions for `exp`, `sin`, and `cos` provided in the Boost interval library, implementing the formula,

$$e^{x+iy} = e^x(\cos y + i \sin y)$$

The function `errorTerm(C z)` constructs a complex interval that contains (at least) all complex values of norm less than or equal to `z`. We use this function to incorporate error terms with an established norm bound. Lastly, we have a simple function `pow(C base, int exp)` that raises a value of type `C` to a natural number

power.

To compute the value of $\log(z)$ for a complex interval representing a value z, we employ the following steps:

1. First, we assert that the interval does not cross our branch cut along the negative x-axis. In fact, we are more draconian and forbid the interval from crossing the real-axis. If an attempt is made to compute such a disallowed logarithm, an exception is thrown and execution is terminated.

2. We choose an arbitrarily rounded center of the interval as a basepoint u, and store it as a complex floating point value (as opposed to an interval).

3. We compute the logarithm, ln_0, of this floating point value using the built-in `log` function from the standard library.

4. Using interval arithmetic, we compute the exponential z_0 of this (using the aforementioned `exp` function).

5. We set $w = \frac{z_0 - z}{z_0}$ and use the following expansion of log about z_0,

$$\log(z) = log(z_0) + log(\frac{z}{z_0}) = log(z_0) + log(1-w) = ln_0 - \sum_{k=1}^{\infty} \frac{w^k}{k}$$
$$= ln_0 - w - \frac{w^2}{2} - \frac{w^3}{3} - \frac{w^4}{4} - \frac{w^5}{5} + \epsilon\left(w^6\right)$$

   This series expansion is valid only inside its radius of convergence, $|w| < 1$. In practice, choosing ln_0 as above will tend to produce a z_0 that is well inside the radius of convergence of the series.

6. Finally, we verify that we obtained the branch desired branch of the logarithm. The above algorithm produces an interval $log(z)$ which contains some value

whose exponential is z. Because the algorithm above produces a logarithm nearby to the built-in double version of log, we might expect the same branch cut to result. Rather than arguing that this is the case, we simply verify at run-time that this is the case: values with positive imaginary component should have logarithms in the open interval $(0, \pi)$ and values with negative imaginary components should have logarithms in the open interval $(-\pi, 0)$. We make it an explicit error to take the logarithm of an interval that overlaps the real-axis.

The implementation of this algorithm is shown in Figure 4.2. We use the `log` function to define `PI` as an interval containing π, as shown. The implementation enforces a conservative requirement that $max(|\mathrm{Re}w|, |\mathrm{Im}w|) < \frac{1}{2}$, which is sufficient to conclude $|w| < 1$. Also, note that we allow the imaginary interval for the computed logarithm to overlap the larger intervals $(-4, 1)$ instead of $(-\pi, 0)$ and $(-1, 4)$ instead of $(0, \pi)$. This allows the computed interval bound for the logarithms imaginary part to include error that puts it outside of the expected range. Since logarithms in the intervals $(-2\pi, -\pi)$ and $(0, \pi)$ are known to have positive imaginary components when exponentiated, there is no potential overlap with other branches when we are taking the logarithm of a number with negative imaginary component. Similary, since logarithms in the intervals $(-\pi, 0)$ and $(\pi, 2\pi)$ are known to have negative imaginary components when exponentiated, there is no potential overlap with other branches when we are taking the logarithm of a number with positive imaginary component.

One note on constants is in order. To avoid reasoning about which real numbers can be represented exactly as literals in decimal form, we will only ever use integer literals to directly construct constants of type `C` in the program. For example, we will encode $\frac{1}{2}$ as `C(1)/C(2)` (and not `C(0.5)`), and we will encode $\frac{i}{3}$ as `C(0,1)/C(3)`. We note that the constructor for `C` accepts either a single parameter specifying the

```
C log(C z)
{
    complex<FP> u(0.5*(z.real().lower() + z.real().upper()),
                  0.5*(z.imag().lower() + z.imag().upper()));
    complex<FP> lnU = log(u);
    C ln0(lnU.real(), lnU.imag());
    C z0 = exp(ln0);
    C w = (z0 - z) / z0;
    if( max(max(w.real().upper(), -w.real().lower()),
            max(w.imag().upper(), -w.imag().lower()))
        >= 0.5)
    {
        throw std::runtime_error("Invalid log");
    }
    C ln = ln0 -            w * (
                C(1)      + w * (
                C(1)/C(2) + w * (
                C(1)/C(3) + w * (
                C(1)/C(4) + w * (
                C(1)/C(5))))))
            + errorTerm(pow(w, 6));
    if(z.imag().lower() > 0.0) {
        if(ln.imag().lower() < -1.0
            || ln.imag().upper() > 4.0)
            throw std::runtime_error("Invalid log");
    }
    else if(z.imag().upper() < 0.0) {
        if(ln.imag().lower() < -4.0
            || ln.imag().upper() > 1.0)
            throw std::runtime_error("Invalid log");
    }
    else
        throw std::runtime_error("Invalid log");
    return ln;
}

    const C PI = log(C(0,1))*C(0,-2);
```

Figure 4.2 : Implementation of the log function

real part of a number or two parameters specifying the real and imaginary parts. Ordinary interval arithmetic will then handle propogating correct interval bounds as they are combined. The compiler will issue an error if an integer literal is too large to be represented exactly.

We have not attempted to perform any formal verifcation of the C++ or Boost numerical libraries. These libraries are known to be expertly designed and they are in wide use. Furthermore, in all cases where we have used the routines to compute known quantities, the exact value has indeed been included in the computed interval.

As a final note, we mention our polcy regarding array subscripting. On occassion in the program supporting this thesis, we will have need to translate subscripted entities in the text into arrays of values in the program. While in the paper we enumerate entities starting from 1, in C++ arrays are indexed from 0 and we adhere to this convention. Thus, for example, the value z_1^+ in the paper corresponds to `ZP[0]` in program, etc.

4.2 Computations on Tori

In order to compute efficiently, we now restrict to the case where Σ is of genus one. Rather than writing A_1 and B_1, we will simply write A and B. The uniformization theorem states that a genus one Riemann surface is obtainable as the quotient of $\mathbf{C}$ by a group of translations generated by $z \to z+1$ and $z \to z+\tau$. We will generally work within a fundamental domain for this quotient that is centered at 0, with our A and B curves running along the sides of this fundamental domain - A from $\frac{-\tau-1}{2}$ to $\frac{-\tau+1}{2}$, and B from $\frac{-\tau-1}{2}$ to $\frac{\tau-1}{2}$.

A surface in Karcher's family admits a reflectional symmetry, and thus our torus is either a rectangle or rhombus, the only types of torus that admit an isometry

of reflection. And as the fixed point set of the Karcher's reflection is disconnected, consisting of a closed inner curve and outer curves that passes through the ends of the surface, we know that we our domain is rectangular and that τ is purely imaginary.

In the remainder of this work, we will be restricting our attention to a single member of the Karcher minimal surface family that has as its domain a rectangular torus with $\tau = 3i$. While not every purely imaginary τ corresponds to a member of Karcher's family, we will verify numerically in a later section (see 4.6) that this choice does.

We may identify functions on Σ with Γ-periodic functions on $\mathbf{C}$. Our approach for numerically constructing such Γ-periodic functions will be to write down a function f with the desired pole structure in a single fundamental domain of Γ, and then write,

$$\langle \mathrm{f} \rangle(\mathrm{z}) = \sum_{\Delta \in \Gamma} \mathrm{f}(\mathrm{z} + \Delta) - \mathrm{g}(\Delta) \tag{4.1}$$

where g, dependent on f, will be chosen to ensure uniform convergence of the series in the fundamental domain (or on any compact set). We will also require of our chosen g that $g(\Delta) \to 0$ as $|\Delta| \to \infty$.

Proposition 4.2.1. *The function* $\langle \mathrm{f} \rangle(\mathrm{z})$ *is well defined up to a constant.*

Proof. Suppose that $g = g_1$ and $g = g_2$ both produce convergence in (4.1). Then,

$$\left(\sum_{\Delta \in \Gamma} f(z + \Delta) - g_1(\Delta) \right) - \left(\sum_{\Delta \in \Gamma} f(z + \Delta) - g_2(\Delta) \right) = \sum_{\Delta \in \Gamma} g_2(\Delta) - g_1(\Delta)$$

which does not depend on z. □

In practice, we will have some way to constrain the constant term, but it is convenient to implement and build upon a basic version first that includes an arbitrary

constant term. For example, if we wish to construct a function with pole structure matching function f on a fundamental domain, and with a zero at point p, we can use,

$$\langle \mathbf{f} \rangle - \langle \mathbf{f} \rangle (\mathrm{p})$$

Or, if we wish to choose the constant term such that the A-period is 0, we can use (since $\int_A dz = 1$),

$$\langle \mathbf{f} \rangle - \int_{\mathrm{A}} \langle \mathbf{f} \rangle$$

Proposition 4.2.2. *Assuming g is chosen to provide convergence, the function $\langle \mathbf{f} \rangle (\mathrm{z})$ as defined in (4.1) is Γ-periodic.*

Proof. We have, for $\Delta' \in \Gamma$,

$$\begin{aligned}
\langle \mathbf{f} \rangle (\mathrm{z} + \Delta') &= \sum_{\Delta \in \Gamma} f(z + (\Delta' + \Delta)) - g(\Delta) = \\
&= \sum_{\Delta'' \in \Gamma} f(z + \Delta'') - g(\Delta'') + \sum_{\Delta'' \in \Gamma} g(\Delta'') - g(\Delta'' - \Delta') \\
&= \langle \mathbf{f} \rangle (\mathrm{z}) + \sum_{\Delta'' \in \Gamma} \mathrm{g}(\Delta'') - \mathrm{g}(\Delta'' - \Delta')
\end{aligned}$$

Here we have introduced $\Delta'' = \Delta' + \Delta$. The latter sum must converge since $\langle \mathbf{f} \rangle$ does. We consider the equivalence relation on Γ of whether elements differ from one another by a multiple of Δ', and we let Γ' consist of one element from each equivalence

class. We can write:

$$\begin{aligned}
&\sum_{\Delta'' \in \Gamma} g(\Delta'') - g(\Delta'' - \Delta') \\
&\quad = \sum_{\Delta \in \Gamma'} \sum_{k=-\infty}^{\infty} g(\Delta + k\Delta') - g(\Delta + (k-1)\Delta') \\
&\quad = 0
\end{aligned}$$

The last equality follows because the series is telescoping and $g(\Delta) \to 0$ as $|\Delta| \to \infty$. □

Our strategy will be to partition Γ into Γ_{in} near the origin and Γ_{out} away from the origin and choose g supported in Γ_{out}. We will then perform the summation over Γ_{in} in (4.1) term for term using interval arithmetic and estimate the value of the sum over Γ_{out}, taking care to obtain valid bounds. We let Γ_{in} consist of a rectangle of complex lattice values of Γ with real part between $-R_K$ and R_K and imaginary part between $-R_J\tau$ and $R_J\tau$. See Figure 4.3. Formally, we let,

$$\Gamma_{in} = \left\{ k + j\tau \,\middle|\, k \in [-R_K, .., R_K], j \in [-R_J, ..., R_J] \right\}$$

Now, let us first consider functions with only simple poles,

$$f(z) = \sum_{k=1}^{N} \frac{r_k}{z - z_k}$$

The residues of such a function must sum to zero, $\sum_{k=1}^{N} r_k = 0$.

We apply the following identity,

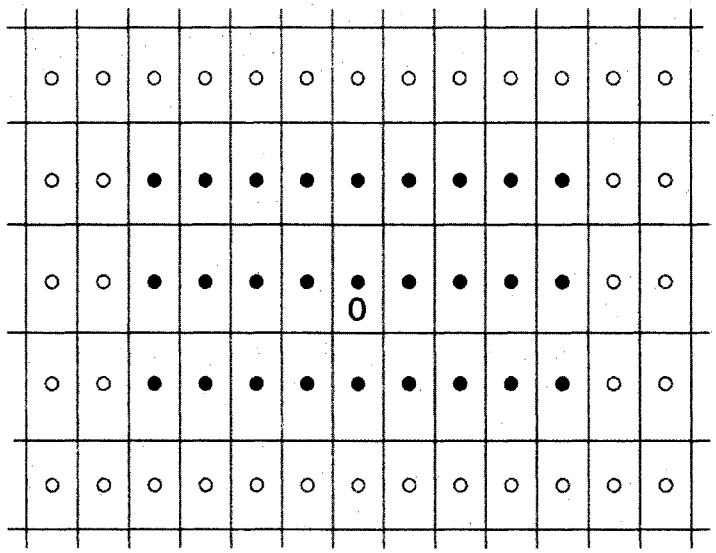

Figure 4.3 : Partition of Γ into Γ_{in} and Γ_{out}.

$$\frac{1}{\Delta - w} = \frac{1}{\Delta}\left(\frac{1}{1 - \frac{w}{\Delta}}\right) \tag{4.2}$$

$$= \frac{1}{\Delta}\left(1 + \frac{w}{\Delta} + (\frac{w}{\Delta})^2 + (\frac{w}{\Delta})^3 + (\frac{w}{\Delta})^4 + (\frac{w}{\Delta})^5 + (\frac{w}{\Delta})^6 + \frac{(\frac{w}{\Delta})^7}{1 - \frac{w}{\Delta}}\right) \tag{4.3}$$

$$= \frac{1}{\Delta} + \frac{w}{\Delta^2} + \frac{w^2}{\Delta^3} + \frac{w^3}{\Delta^4} + \frac{w^4}{\Delta^5} + \frac{w^5}{\Delta^6} + \frac{w^6}{\Delta^7} + \frac{w^7}{\Delta^8 - w\Delta^7} \tag{4.4}$$

Here we have chosen to stop the expansion at the $\frac{1}{\Delta^8}$ term somewhat arbitrarily, as this will provide a sufficiently precise approximation for our needs in what follows. Multiplying by r_k and substituting $w = z_k - z$, we find that,

$$f(z+\Delta) = \sum_{k=1}^{N} \frac{r_k}{\Delta - (z_k - z)}$$

$$= \frac{\sum_{k=1}^{N} r_k}{\Delta} + \frac{\sum_{k=1}^{N} r_k (z_k - z)}{\Delta^2} + \ldots + \sum_{k=1}^{N} \frac{r_k (z_k - z)^7}{\Delta^8 - (z_k - z)\Delta^7}$$

As $\sum_{k=1}^{N} r_k = \sum_{k=1}^{N} r_k z = 0$, the coefficient on the $\frac{1}{\Delta}$ term is 0, and the coefficient on the $\frac{1}{\Delta^2}$ term does not depend on z. As the rest of the terms $\frac{1}{\Delta^n}$, $n \geq 3$, converge already, we may obtain the overall convergence we desire by choosing,

$$g(\Delta) = \begin{cases} \frac{\sum_{k=1}^{N} r_k z_k}{\Delta^2}, & \Delta \in \Gamma_{out} \\ 0, & \Delta \in \Gamma_{in} \end{cases} \tag{4.5}$$

We notice that $\langle \mathbf{f} \rangle$ can be computed as a sum of terms, each of which only depends on the location and residue of a single pole of f. It will be convenient to generalize our notation based on this observation to let, for $f(z) = \frac{1}{z - z_0}$,

$$\langle \mathbf{f} \rangle(z) = \sum_{\Delta \in \Gamma_{in}} \frac{1}{\Delta + z - z_0} + \sum_{\Delta \in \Gamma_{out}} \left(\frac{1}{\Delta + z - z_0} - \frac{1}{\Delta} - \frac{(z_0 - z)}{\Delta^2} \right)$$

We know that $\langle \mathbf{f} \rangle$ in this case will not be a Γ-periodic function, as a meromorphic function with single pole does not exist on a compact Riemann surface. But our generalized notation is justified by the fact that, when $\sum_{k=1}^{N} r_k = 0$, we have,

$$\langle \sum_{k=1}^{N} \frac{r_k}{z - z_k} \rangle = \sum_{k=1}^{N} r_k \langle \frac{1}{z - z_k} \rangle$$

We now compute,

$$\begin{aligned} &\frac{1}{\Delta - (z_0 - z)} - \frac{1}{\Delta} - \frac{z_0 - z}{\Delta^2} \\ &= \frac{(z_0 - z)^2}{\Delta^3} + \frac{(z_0 - z)^3}{\Delta^4} + \frac{(z_0 - z)^4}{\Delta^5} \\ &\quad + \frac{(z_0 - z)^5}{\Delta^6} + \frac{(z_0 - z)^6}{\Delta^7} + \frac{(z_0 - z)^7}{\Delta^8 - (z_0 - z)\Delta^7} \end{aligned}$$

For $\tau \leq 3$, we bound,

$$|z_0 - z| \leq \sqrt{1 + \tau^2} < 3.2$$
$$|\Delta^8 - (z_0 - z)\Delta^7| \geq |\Delta|^8 (1 - |\frac{z_0 - z}{\Delta}|) \geq |\Delta|^8 (1 - \frac{4}{101}) \geq 0.9 |\Delta|^8$$

We also note that for odd n we have, by symmetry,

$$\sum_{\Delta \in \Gamma_{out}} \frac{1}{\Delta^n} = \sum_{\Delta \in \Gamma_{out}} \frac{1}{(-\Delta)^n} = -\left(\sum_{\Delta \in \Gamma_{out}} \frac{1}{\Delta^n} \right) = 0$$

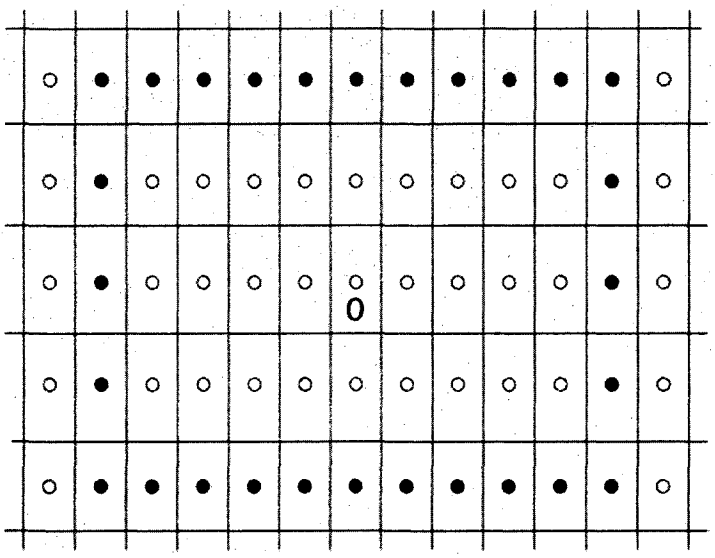

Figure 4.4 : Example of concentric rectangle comprising the elements of Γ_n

We introduce the notation $\epsilon(z)$ to refer to some unnamed complex error term w with $|w| \leq |z|$, and put the preceding together as,

$$\sum_{\Delta \in \Gamma_{out}} \frac{1}{\Delta + z - z_0} - \frac{1}{\Delta} - \frac{(z_0 - z)}{\Delta^2} = \sum_{\Delta \in \Gamma_{out}} \frac{(z_0 - z)^3}{\Delta^4} + \frac{(z_0 - z)^5}{\Delta^6} + \epsilon\left(\frac{3818}{\Delta^8}\right)$$

We bound the error term summed over Γ_{out} by summing over concentric rectangles (see Figure 4.4). For this estimate we assume $|\tau| \geq 1$ (which is conservative as we are

interested in the case $\tau = 3i$). Formally, we partition $\Gamma_{out} = \bigcup_{n\geq 1} \Gamma_n$, where

$$B_n = \left\{k + j\tau \,\middle|\, k \in [-R_K - n, .., R_K + n], j \in [-R_J - n, ..., R_J + n]\right\}$$
$$\Gamma_n = B_n - B_{n-1}$$

For $|\tau| \geq 1$, we have $R_K \geq R_J$ and thus the number of elements in Γ_n is smaller than $(2(R_K + n) + 1)^2 - (2(R_K + n) - 1)^2 = 8(R_K + n)$ elements, the number of elements in concentric squares if $R_K = R_J$. Furthermore, with $|\tau| \geq 1$, we have the bound $|\Delta| \leq R_K + n$ for each $\Delta \in \Gamma_n$. Therefore,

$$\begin{aligned}\sum_{\Delta\in\Gamma_{out}} \epsilon\left(\frac{1}{\Delta^8}\right) &\leq \sum_{R=R_K+1}^{\infty} 8R\epsilon\left(\frac{1}{R^8}\right) \\ &= \sum_{R=R_K+1}^{\infty} \epsilon\left(\frac{8}{R^7}\right)\end{aligned}$$

Since $\frac{1}{x^7}$ is a decreasing function of x, we have,

$$\frac{1}{R^7} < \int_{x=R-1}^{R} \frac{dx}{x^7}$$

$$\sum_{\Delta\in\Gamma_{out}} \epsilon\left(\frac{1}{\Delta^8}\right) = \sum_{R=R_K+1}^{\infty} \epsilon\left(\frac{8}{R^7}\right) = \epsilon\left(\int_{x=R_K}^{\infty} \frac{8}{x^7}\right) = \epsilon\left(\frac{4}{3R_K^6}\right) \tag{4.6}$$

Putting this together, we have proven the first main proposition of this section,

Proposition 4.2.3. *For $f(z) = \frac{1}{z-z_0}$, with g as in (4.5), we have,*

$$\begin{aligned}\langle \mathbf{f} \rangle(z) = &\left(\sum_{\Delta \in \Gamma_{in}} \frac{1}{\Delta + z - z_0}\right) \\ &- (z-z_0)^3 \left(\sum_{\Delta \in \Gamma_{out}} \frac{1}{\Delta^4}\right) - (z-z_0)^5 \left(\sum_{\Delta \in \Gamma_{out}} \frac{1}{\Delta^6}\right) \\ &+ \epsilon \left(\frac{5091}{R_K^6}\right)\end{aligned}$$

This proposition is translated to the implementation of `evaluateSimplePole` shown in Figure 4.5 as a means for computing $\langle \mathbf{f} \rangle(z)$ in the case of a simple pole, in the sense that the implemented source code is a direct idiomatic translation of the formula of the proposition. The implementation relies on helper functions `log`, `pow`, and `errorTerm`, described previously, as well as correct implementations of `sumDeltaInv4thOut` and `sumDeltaInv6thOut` to compute $\sum_{\Delta \in \Gamma_{out}} \frac{1}{\Delta^4}$ and $\sum_{\Delta \in \Gamma_{out}} \frac{1}{\Delta^6}$, respectively. We will return in the next section to the problem of estimating those series to provide these implementations.

Proposition 4.2.4.

$$\begin{aligned}\int_A \langle \mathbf{f} \rangle(z) = &\left(\sum_{j=-R_J}^{R_J} \log(R_K + \frac{1}{2} + (j - \frac{1}{2})\tau - z_0) - \log(-R_K - \frac{1}{2} + (j - \frac{1}{2})\tau - z_0)\right) \\ &+ \frac{1}{4}\left((-\frac{\tau}{2} - \frac{1}{2} - z_0)^4 - (-\frac{\tau}{2} + \frac{1}{2} - z_0)^4\right)\left(\sum_{\Delta \in \Gamma_{out}} \frac{1}{\Delta^4}\right) \\ &+ \frac{1}{6}\left((-\frac{\tau}{2} - \frac{1}{2} - z_0)^6 - (-\frac{\tau}{2} + \frac{1}{2} - z_0)^6\right)\left(\sum_{\Delta \in \Gamma_{out}} \frac{1}{\Delta^6}\right) \\ &+ \epsilon\left(\frac{5091}{R_K^6}\right)\end{aligned}$$

```
C evaluateSimplePole(C z0, C z)
{
    C sum(0);
    for(int j= -RJ; j <= RJ; ++j)
        for(int k = -RK; k <= RK; ++k)
        {
            C delta = C(k) + C(j) * tau;
            sum += C(1) / (delta + z - z0);
        }

    sum += -pow(z - z0, 3) * sumDeltaInv4thOut()
        -  pow(z - z0, 5) * sumDeltaInv6thOut()
        +  errorTerm(C(5091) / pow(C(RK), 6));
    return sum;
}
```

Figure 4.5 : Source for `evaluateSimplePole` corresponding to Proposition 4.2.3

Proof. This proposition follows from integrating the formula of Proposition 4.2.3 from $\frac{-1-\tau}{2}$ to $\frac{1-\tau}{2}$, which is our chosen representative for the path class of A. Because the branch cut for *log* is taken along the negative real axis, none of the paths of integration intersect the branch cut. □

This proposition is translated to the implementation of `integrateSimplePoleAlongA` shown in Figure 4.6.

We now go through similar steps for a second order pole,

$$f(z) = \frac{1}{(z - z_0)^2}$$

We differentiate a formula similar to (4.4), but with one fewer term in the expansion, to obtain a similar expression for a second order pole:

```
C integrateSimplePoleAlongA(C z0)
{
    C sum(0);
    for(int j= -RJ; j <= RJ; ++j)
    {
        sum += log( C(RK) + C(1)/C(2)
                    + (C(j) - C(1)/C(2))*tau - z0)
            -  log(-C(RK) - C(1)/C(2)
                    + (C(j) - C(1)/C(2))*tau - z0);
    }

    sum += C(1)/C(4)
            * ( pow(-C(1)/C(2) - tau/C(2) - z0, 4)
            -pow( C(1)/C(2) - tau/C(2) - z0, 4))
            * sumDeltaInv4thOut();
        + C(1)/C(6)
            * ( pow(-C(1)/C(2) - tau/C(2) - z0, 6)
            -pow( C(1)/C(2) - tau/C(2) - z0, 6))
            * sumDeltaInv6thOut();
        + errorTerm(C(5091) / pow(C(RK), 6));

    return sum;
}
```

Figure 4.6 : Source for `evaluateSimplePole` corresponding to Proposition 4.2.4

$$\begin{aligned}\frac{1}{(\Delta-w)^2} &= -\frac{d}{d\Delta}\left(\frac{1}{\Delta-w}\right) \\ &= -\frac{d}{d\Delta}\left(\frac{1}{\Delta}+\frac{w}{\Delta^2}+\frac{w^2}{\Delta^3}+\frac{w^3}{\Delta^4}+\frac{w^4}{\Delta^5}+\frac{w^5}{\Delta^6}+\frac{w^6}{\Delta^7-w\Delta^6}\right) \\ &= \frac{1}{\Delta^2}+\frac{2w}{\Delta^3}+\frac{3w^2}{\Delta^4}+\frac{4w^3}{\Delta^5}+\frac{5w^4}{\Delta^6}+\frac{6w^5}{\Delta^7}+\frac{7w^6\Delta^6-6w^7\Delta^5}{(\Delta^7-w\Delta^6)^2}\end{aligned}$$

Substituting $w = z_0 - z$, we find,

$$f(\Delta+z) = \frac{1}{(\Delta-(z_0-z))^2} = \frac{1}{\Delta^2}+\frac{2(z_0-z)}{\Delta^3}+\ldots+\frac{7(z_0-z)^6\Delta^6-6(z_0-z)^7\Delta^5}{(\Delta^7-(z_0-z)\Delta^6)^2}$$

In this case again, there is no $\frac{1}{\Delta}$ term and the $\frac{1}{\Delta^2}$ term does not depend on z and can thus be incorporated into g. We are thus led to define g as:

$$g(\Delta) = \begin{cases} \frac{1}{\Delta^2}, & \Delta \in \Gamma_{out} \\ 0, & \Delta \in \Gamma_{in} \end{cases} \tag{4.7}$$

For this f with a second order pole we thus have,

$$\langle \mathbf{f} \rangle(z) = \sum_{\Delta\in\Gamma_{\text{in}}} \frac{1}{(\Delta+z-z_0)^2} + \sum_{\Delta\in\Gamma_{\text{out}}} \left(\frac{1}{(\Delta+z-z_0)^2}-\frac{1}{\Delta^2}\right)$$

We compute,

$$\frac{1}{(\Delta-(z_0-z))^2}-\frac{1}{\Delta^2}= \frac{2(z_0-z)}{\Delta^3}+\frac{3(z_0-z)^2}{\Delta^4}+\frac{4(z_0-z)^3}{\Delta^5}+\frac{5(z_0-z)^4}{\Delta^6} +\frac{6(z_0-z)^5}{\Delta^7}+\frac{7(z_0-z)^6\Delta^6-6(z_0-z)^7\Delta^5}{(\Delta^7-(z_0-z)\Delta^6)^2}$$

Analogously to the simple pole case, we have for $|\tau|\leq 3$ and $|\Delta|\geq 101$,

$$|7(z_0-z)^6\Delta^6-6(z_0-z)^7\Delta^5|\leq|\Delta^6|(7(3.2^6)+6\frac{3.2^7}{101})=7721|\Delta^6|$$
$$|\Delta^7-(z_0-z)\Delta^6|=|\Delta^7|(1-\frac{z_0-z}{\Delta})\geq|\Delta^7|(1-\frac{3.2}{101})\geq .96|\Delta|^7$$

With the odd powers of $\frac{1}{\Delta}$ summing to 0, by symmetry, we have,

$$\sum_{\Delta\in\Gamma_{out}}\left(\frac{1}{(\Delta+z-z_0)^2}-\frac{1}{\Delta^2}\right)=\sum_{\Delta\in\Gamma_{out}}\frac{3(z_0-z)^2}{\Delta^4}+\frac{5(z_0-z)^4}{\Delta^6}+\epsilon\left(\frac{8378}{\Delta^8}\right)$$

Using (4.6) to bound the summed error terms, we prove,

Proposition 4.2.5. *For $f(z)=\frac{1}{(z-z_0)^2}$ with g as in (4.7),*

$$\langle\mathbf{f}\rangle(z)=\left(\sum_{\Delta\in\Gamma_{in}}\frac{1}{(\Delta+z-z_0)^2}\right) +3(z-z_0)^2\left(\sum_{\Delta\in\Gamma_{out}}\frac{1}{\Delta^4}\right)+5(z-z_0)^4\left(\sum_{\Delta\in\Gamma_{out}}\frac{1}{\Delta^6}\right) +\epsilon\left(\frac{11171}{R_K^6}\right)$$

This proposition is translated to the implementation of `evaluateSecondOrderPole` shown in Figure 4.7.

```
C evaluateSecondOrderPole(C z0, C z)
{
    C sum(0);
    for(int j= -RJ; j <= RJ; ++j)
        for(int k = -RK; k <= RK; ++k)
        {
            C delta = C(k) + C(j) * tau;
            sum += C(1) / pow(delta + z - z0, 2);

            if(j!=0 || k!=0)
                sum -= C(1) / pow(delta, 2);
        }

    sum += C(3) * pow(z - z0, 2) * sumDeltaInv4thOut()
        + C(5) * pow(z - z0, 4) * sumDeltaInv6thOut()
        + errorTerm(C(11171) / pow(C(RK), 6));
    return sum;
}
```

Figure 4.7 : Source for `evaluateSecondOrderPole` corresponding to Proposition 4.2.5

Proposition 4.2.6. *For* $f(z) = \frac{1}{(z-z_0)^2}$,

$$\begin{aligned}\int_A \langle \mathbf{f} \rangle(\mathbf{z}) = &\left(\sum_{j=-R_J}^{R_J} \frac{1}{-R_K - \frac{1}{2} + (j - \frac{1}{2})\tau - z_0} - \frac{1}{R_K + \frac{1}{2} + (j - \frac{1}{2})\tau - z_0} \right) \\ &+ \left((\frac{1}{2} - \frac{\tau}{2} - z_0)^3 - (-\frac{1}{2} - \frac{\tau}{2} - z_0)^3 \right) \left(\sum_{\Delta \in \Gamma_{out}} \frac{1}{\Delta^4} \right) \\ &+ \left((\frac{1}{2} - \frac{\tau}{2} - z_0)^5 - (-\frac{1}{2} - \frac{\tau}{2} - z_0)^5 \right) \left(\sum_{\Delta \in \Gamma_{out}} \frac{1}{\Delta^6} \right) \\ &+ \epsilon \left(\frac{11171}{R_K^6} \right)\end{aligned}$$

Proof. This proposition follows from integrating the formula of Proposition 4.2.6 from $\frac{-1-\tau}{2}$ to $\frac{1-\tau}{2}$, which is our chosen representative for the path class of A. □

This proposition is translated to the implementation of `integrateSecondOrderPoleAlongA` shown in Figure 4.8.

Proposition 4.2.7. *For* $f(z) = \frac{1}{(z-z_0)^2}$,

$$\begin{aligned}\int_B \langle \mathbf{f} \rangle(\mathbf{z}) = &\left(\sum_{k=-R_K}^{R_K} \frac{1}{k - \frac{1}{2} + (-R_J - \frac{1}{2})\tau - z_0} - \frac{1}{k - \frac{1}{2} + (R_J + \frac{1}{2})\tau - z_0} \right) \\ &+ \left((-\frac{1}{2} + \frac{\tau}{2} - z_0)^3 - (-\frac{1}{2} - \frac{\tau}{2} - z_0)^3 \right) \left(\sum_{\Delta \in \Gamma_{out}} \frac{1}{\Delta^4} \right) \\ &+ \left((-\frac{1}{2} + \frac{\tau}{2} - z_0)^5 - (-\frac{1}{2} - \frac{\tau}{2} - z_0)^5 \right) \left(\sum_{\Delta \in \Gamma_{out}} \frac{1}{\Delta^6} \right) \\ &+ \epsilon \left(\frac{33513}{R_K^6} \right)\end{aligned}$$

Proof. This proposition follows from integrating the formula of Proposition 4.2.7 from $\frac{-1-\tau}{2}$ to $\frac{-1+\tau}{2}$, which is our chosen representative for the path class of B. □

This proposition is translated to the implementation of

```
C integrateSecondOrderPoleAlongA(C z0)
{
    C sum(0);
    for(int j= -RJ; j <= RJ; ++j)
    {
        sum += C(1) / (-C(RK) - C(1)/C(2)
                       + (C(j) - C(1)/C(2))*tau - z0)
            - C(1) / ( C(RK) + C(1)/C(2)
                       + (C(j) - C(1)/C(2))*tau - z0);
    }

    sum += ( pow(C(1)/C(2) - tau/C(2) - z0, 3) -
             pow(-C(1)/C(2) - tau/C(2) - z0, 3) )
        * sumDeltaInv4thOut()
        + ( pow(C(1)/C(2) - tau/C(2) - z0, 5)
            -pow(-C(1)/C(2) - tau/C(2) - z0, 5) )
        * sumDeltaInv6thOut()
        + errorTerm(C(11171) / pow(C(RK), 6));

    return sum;
}
```

Figure 4.8 : Source for `integrateSecondOrderPoleAlongA` corresponding to Proposition 4.2.6

```
C integrateSecondOrderPoleAlongB(C z0)
{
    C sum(0);
    for(int k= -RK; k <= RK; ++k)
    {
        sum += C(1) / ( C(k) - C(1)/C(2)
                        + (-C(RJ) - C(1)/C(2))*tau - z0)
            - C(1) / ( C(k) - C(1)/C(2)
                        + (C(RJ) + C(1)/C(2))*tau - z0);
    }

    sum += ( pow(-C(1)/C(2) + tau/C(2) - z0, 3)
            -pow(-C(1)/C(2) - tau/C(2) - z0, 3) )
            * sumDeltaInv4thOut()
        + ( pow(-C(1)/C(2) + tau/C(2) - z0, 5)
            -pow(-C(1)/C(2) - tau/C(2) - z0, 5) )
            * sumDeltaInv6thOut()
        + errorTerm(C(33513) / pow(C(RK), 6));

    return sum;
}
```

Figure 4.9 : Source for `integrateSecondOrderPoleAlongB` corresponding to Proposition 4.2.7

`integrateSecondOrderPoleAlongB` shown in Figure 4.9.

Proposition 4.2.8. *For* $f(z) = \frac{1}{(z-z_0)^2}$, *and* γ *a curve on* Σ *with endpoints corre-*

sponding to $a, b \in C$,

$$\begin{aligned}\int_\gamma \langle \mathbf{f} \rangle(z) = &\left(\sum_{\Delta \in \Gamma_{in}} \frac{1}{\Delta + a - z_0} - \frac{1}{\Delta + b - z_0} \right) \\ &+ \left((b - z_0)^3 - (a - z_0)^3 \right) \left(\sum_{\Delta \in \Gamma_{out}} \frac{1}{\Delta^4} \right) \\ &+ \left((b - z_0)^5 - (a - z_0)^5 \right) \left(\sum_{\Delta \in \Gamma_{out}} \frac{1}{\Delta^6} \right) \\ &+ \epsilon \left(\frac{11171(b-a)}{R_K^6} \right)\end{aligned}$$

Proof. This proposition follows from integrating the formula of Proposition 4.2.8 from a to b. □

This proposition is translated to the implementation of `integrateSecondOrderPoleAlongPath` shown in Figure 4.10.

We now introduce some additional helper functions for the common case where the function we are evaluating or integrating is known to have vanishing A-period (see Figure 4.11).

4.3 Formulae for $\sum_{\Delta \in \Gamma_{out}} \frac{1}{\Delta^4}$ and $\sum_{\Delta \in \Gamma_{out}} \frac{1}{\Delta^6}$

Now we seek approximations for $\sum_{\Delta \in \Gamma_{out}} \frac{1}{\Delta^4}$ and $\sum_{\Delta \in \Gamma_{out}} \frac{1}{\Delta^6}$. We define,

$$P_n(\Delta) = \int_{\Delta - \frac{1}{2}}^{\Delta + \frac{1}{2}} \frac{dz}{z^n} = \frac{1}{n-1} \left(\frac{1}{(\Delta - \frac{1}{2})^{n-1}} - \frac{1}{(\Delta + \frac{1}{2})^{n-1}} \right)$$

```
C integrateSecondOrderPoleAlongPath(C z0, C a, C b)
{
    C sum(0);
    for(int j= -RJ; j <= RJ; ++j)
        for(int k = -RK; k <= RK; ++k)
        {
            C delta = C(k) + C(j) * tau;
            sum += C(1) / (delta + a - z0)
                -  C(1) / (delta + b - z0);
        }

    sum += ( pow(b - z0, 3) - pow(a - z0, 3) )
        * sumDeltaInv4thOut()
        +  ( pow(b - z0, 5) - pow(a - z0, 5) )
        * sumDeltaInv6thOut()
        + errorTerm(C(11171) * (b - a) / pow(C(RK), 6));

    return sum;
}
```

Figure 4.10 : Source for `integrateSecondOrderPoleAlongPath` corresponding to Proposition 4.2.8

```
C evaluateSimplePoleZeroAPeriod(C z0, C z)
{
    return evaluateSimplePole(z0, z)
            - integrateSimplePoleAlongA(z0);
}

C evaluateSecondOrderPoleZeroAPeriod(C z0, C z)
{
    return evaluateSecondOrderPole(z0, z)
            - integrateSecondOrderPoleAlongA(z0);
}

C integrateSecondOrderPoleZeroAPeriodAlongB(C z0)
{
    return integrateSecondOrderPoleAlongB(z0)
            - tau * integrateSecondOrderPoleAlongA(z0);
}

C integrateSecondOrderPoleZeroAPeriod(C z0, C a, C b)
{
    return integrateSecondOrderPoleAlongPath(z0, a, b)
            - (b - a) * integrateSecondOrderPoleAlongA(z0);
}
```

Figure 4.11 : Source for some helper functions

For any j and $n > 2$, we have,

$$\sum_{k=k_0}^{\infty} P_n(k+j\tau) = \frac{1}{(n-1)(k_0+j\tau-\frac{1}{2})^{n-1}} \qquad k_0 > 0 \qquad (4.8)$$

$$\sum_{k=-\infty}^{k_1} P_n(k+j\tau) = \frac{-1}{(n-1)(k_1+j\tau+\frac{1}{2})^{n-1}} \qquad k_1 < 0 \qquad (4.9)$$

$$\sum_{k=-\infty}^{\infty} P_n(k+j\tau) = 0 \qquad j \neq 0 \qquad (4.10)$$

These results follow from the fact that the series are, by design, telescoping with terms approach zero for $n > 1$. We now show that $P_4(\Delta)$ is a 6th order approximation to $\frac{1}{\Delta^4}$, and will soon thereafter see that $P_4(\Delta) - \frac{5}{6}P_6(\Delta)$ is an approximation to 8th order.

To begin, note that,

$$P_4(\Delta) = \frac{1}{3}\left(\frac{1}{(\Delta-\frac{1}{2})^3} - \frac{1}{(\Delta+\frac{1}{2})^3}\right) = \frac{\Delta^2+\frac{1}{12}}{(\Delta^2-\frac{1}{4})^3}$$

so that,

$$P_4(\Delta) - \frac{1}{\Delta^4} = \frac{\frac{5}{6}\Delta^4 - \frac{3}{16}\Delta^2 + \frac{1}{64}}{\Delta^4(\Delta^2-\frac{1}{4})^3} = \frac{5}{6}\frac{1}{(\Delta^2-\frac{1}{4})^3} + \frac{-\frac{3}{16}\Delta^2+\frac{1}{64}}{\Delta^4(\Delta^2-\frac{1}{4})^3}$$

We bound,

$$|-\frac{3}{16}\Delta^2 + \frac{1}{64}| \leq |\Delta|^2(\frac{3}{16} + \frac{1}{64(101)^2}) < 0.2|\Delta|^2$$

$$|\Delta^2 - \frac{1}{4}| \geq |\Delta|^2(1 - \frac{1}{4(101)^2}) > 0.9|\Delta|^2$$

$$|\frac{-\frac{3}{16}\Delta^2+\frac{1}{64}}{\Delta^4(\Delta^2-\frac{1}{4})^3}| < \frac{0.2|\Delta|^2}{0.9^3|\Delta|^{10}} < \frac{0.3}{|\Delta|^8}$$

Then we compute,

$$\frac{1}{(\Delta^2-\frac{1}{4})^3}-\frac{1}{\Delta^6}=\frac{\frac{3}{4}\Delta^4-\frac{3}{16}\Delta^2+\frac{1}{64}}{\Delta^6(\Delta^2-\frac{1}{4})^3}$$

And again we bound,

$$|\frac{3}{4}\Delta^4-\frac{3}{16}\Delta^2+\frac{1}{64}|\leq|\Delta|^4(\frac{3}{4}-\frac{3}{16(101)^2}+\frac{1}{64(101)^4})<0.8|\Delta|^4$$
$$|\Delta^2-\frac{1}{4}|\geq|\Delta^2|(1-\frac{1}{4(101)^2})>0.9|\Delta^2|$$
$$|\frac{\frac{3}{4}\Delta^4-\frac{3}{16}\Delta^2+\frac{1}{64}}{\Delta^6(\Delta^2-\frac{1}{4})^3}|<\frac{1.1}{|\Delta|^8}$$

Thus,

$$\frac{1}{\Delta^4}=P_4(\Delta)-\frac{5}{6}\left(\frac{1}{\Delta^6}+\epsilon\left(\frac{1.1}{|\Delta|^8}\right)\right)+\epsilon\left(\frac{0.3}{|\Delta|^8}\right) \tag{4.11}$$
$$=P_4(\Delta)-\frac{5}{6}\frac{1}{\Delta^6}+\epsilon\left(\frac{1.3}{|\Delta|^8}\right) \tag{4.12}$$

We next show that $P_6(\Delta)$ is an 8th order approximation of $\frac{1}{\Delta^6}$, and substitute this approximation back into the preceding equation for an 8th order approximation to $\frac{1}{\Delta^4}$.

$$P_6(\Delta)=\frac{\Delta^4+\frac{1}{2}\Delta^2+\frac{1}{90}}{(\Delta^2-\frac{1}{4})^5}$$

$$P_6(\Delta)-\frac{1}{\Delta^6}=\frac{\frac{1}{2}\Delta^8+\frac{1}{90}\Delta^6+\frac{5}{4}\Delta^8-\frac{10}{16}\Delta^6+\frac{10}{32}\Delta^4-\frac{5}{64}\Delta^2+\frac{1}{128}}{\Delta^6(\Delta^2-\frac{1}{4})^5}$$

We bound,

$$\left|\frac{1}{2}\Delta^8+\frac{1}{90}\Delta^6+\frac{5}{4}\Delta^8-\frac{10}{16}\Delta^6+\frac{10}{32}\Delta^4-\frac{5}{64}\Delta^2+\frac{1}{128}\right|$$
$$<|\Delta|^8\left(1.75+\frac{1}{90(101)^2}+\frac{10}{16(101)^2}+\frac{10}{32(101)^4}+\frac{5}{64(101)^6}+\frac{1}{128(101)^8}\right)<1.8|\Delta|^8$$
$$\frac{\frac{1}{4}\Delta^8+\frac{1}{90}\Delta^6+\frac{5}{4}\Delta^8-\frac{10}{16}\Delta^6+\frac{10}{32}\Delta^4-\frac{5}{64}\Delta^2+\frac{1}{128}}{\Delta^6(\Delta^2-\frac{1}{4})^5}<\frac{1.8|\Delta|^8}{0.9^5|\Delta|^{16}}<\frac{3.1}{|\Delta|^8}$$

We collect these results as:

Proposition 4.3.1.

$$\frac{1}{\Delta^6}=P_6(\Delta)+\epsilon\left(\frac{3.1}{|\Delta|^8}\right)$$
$$\frac{1}{\Delta^4}=P_4(\Delta)-\frac{5}{6}P_6(\Delta)+\epsilon\left(\frac{3.7}{|\Delta|^8}\right)$$

Proof. The formula for $\frac{1}{\Delta^6}$ is immediate from the preceding paragraph. For the second formula, we substitute the first into (4.12):

$$\begin{aligned}\frac{1}{\Delta^4}&=P_4(\Delta)-\frac{5}{6}\left(P_6(\Delta)+\epsilon\left(\frac{3.1}{|\Delta|^8}\right)\right)+\epsilon\left(\frac{1.3}{|\Delta|^8}\right)\\&=P_4(\Delta)-\frac{5}{6}P_6(\Delta)+\epsilon\left(\frac{3.9}{|\Delta|^8}\right)\end{aligned}$$

□

Finally, we use these expressions to find,

$$\sum_{\Delta\in\Gamma_{out}}\frac{1}{\Delta^4}=\sum_{j=-\infty}^{-R_J-1}\sum_{k=-\infty}^{\infty}P_4(k+j\tau)-\frac{5}{6}P_6(k+j\tau)$$
$$+\sum_{j=-R_J}^{R_J}\sum_{k=-\infty}^{-R_K-1}P_4(k+j\tau)-\frac{5}{6}P_6(k+j\tau)$$
$$+\sum_{j=-R_J}^{R_J}\sum_{k=R_K+1}^{\infty}P_4(k+j\tau)-\frac{5}{6}P_6(k+j\tau)$$
$$+\sum_{j=R_J+1}^{\infty}\sum_{k=-\infty}^{\infty}P_4(k+j\tau)-\frac{5}{6}P_6(k+j\tau)$$
$$+\sum_{\Delta\in\Gamma_{out}}\epsilon\left(\frac{3.9}{|\Delta|^8}\right)$$

The first and fourth terms are 0, by (4.10), and the second and third terms are given by (4.8) and (4.9). Putting these elements together, and reusing (4.6), we have,

$$\sum_{\Delta\in\Gamma_{out}}\frac{1}{\Delta^4}=\Big(\sum_{j=-R_J}^{R_J}\frac{-1}{3(-R_K-1+j\tau+\frac{1}{2})^3}-\frac{5}{6}\frac{-1}{5(-R_K-1+j\tau+\frac{1}{2})^5}$$
$$+\frac{1}{3(R_K+1+j\tau-\frac{1}{2})^3}-\frac{5}{6}\frac{1}{5(R_K+1+j\tau-\frac{1}{2})^5}\Big)$$
$$+\epsilon\left(\frac{5.2}{R_K^6}\right)$$
$$=\Big(\sum_{j=-R_J}^{R_J}\frac{-1}{3(-R_K-\frac{1}{2}+j\tau)^3}+\frac{1}{6(-R_K-\frac{1}{2}+j\tau)^5}$$
$$+\frac{1}{3(R_K+\frac{1}{2}+j\tau)^3}-\frac{1}{6(R_K+\frac{1}{2}+j\tau)^5}\Big)$$
$$+\epsilon\left(\frac{5.2}{R_K^6}\right)$$

Similarly, we compute,

$$\begin{aligned}\sum_{\Delta\in\Gamma_{out}}\frac{1}{\Delta^6} &= \sum_{j=-\infty}^{-R_J-1}\sum_{k=-\infty}^{\infty} P_6(k+j\tau)\\ &+ \sum_{j=-R_J}^{R_J}\sum_{k=-\infty}^{-R_K-1} P_6(k+j\tau)\\ &+ \sum_{j=-R_J}^{R_J}\sum_{k=R_K+1}^{\infty} P_6(k+j\tau)\\ &+ \sum_{j=R_J+1}^{\infty}\sum_{k=-\infty}^{\infty} P_6(k+j\tau)\\ &+ \sum_{\Delta\in\Gamma_{out}}\epsilon\left(\frac{3.1}{|\Delta|^8}\right)\end{aligned}$$

Again the first and fourth terms are 0, and the second and third are given by (4.8) and (4.9). We again use (4.6) to sum the error term. We collect these results as,

Proposition 4.3.2.

$$\begin{aligned}\sum_{\Delta\in\Gamma_{out}}\frac{1}{\Delta^4} &= \Big(\sum_{j=-R_J}^{R_J}\frac{-1}{3(-R_K-\frac{1}{2}+j\tau)^3}+\frac{1}{6(-R_K-\frac{1}{2}+j\tau)^5}\\ &\quad+\frac{1}{3(R_K+\frac{1}{2}+j\tau)^3}-\frac{1}{6(R_K+\frac{1}{2}+j\tau)^5}\Big)\\ &\quad+\epsilon\left(\frac{5.2}{R_K^6}\right)\\ \sum_{\Delta\in\Gamma_{out}}\frac{1}{\Delta^6} &= \Big(\sum_{j=-R_J}^{R_J}\frac{1}{5(-R_K-1+j\tau+\frac{1}{2})^5}-\frac{1}{5(R_K+1+j\tau-\frac{1}{2})^5}\Big)+\ \epsilon\left(\frac{4.2}{R_K^6}\right)\end{aligned}$$

This proposition is translated to the implementations of `sumDeltaInv4thOut` and

`sumDeltaInv6thOut` shown in Figure 4.12.

4.4 Computations for Karcher's surface

In this section we apply the machinery developed in the previous section to the implementation of numerical methods that demonstrate the independence of the quadratic differentials listed in Proposition 3.9.2 for a member of Karcher's family of minimal surfaces.

Our approach to the independence of the differentials is based on the following proposition, noting that on a torus, the poles and zeroes of η must be equal in number, so that $n_e = n_z$; in this section we will only use n_z. Also, as our surface is genus one, we drop subscripts from the forms that were indexed by 1, ..., n_g, such as P_B.

Proposition 4.4.1. *Let Φ_1, ..., Φ_n denote the quadratic differentials of Proposition 3.9.2. These differentials are linearly independent over the reals if and only if the following matrix is full rank:*

$$\begin{bmatrix} a_{1,1} & \cdots & a_{1,n_z} & b_{1,1} & \cdots & b_{1,n_z} & c_{1,1} & \cdots & c_{1,n_z} & d_{1,1} & \cdots & d_{1,n_z} & e_1 \\ \vdots & & & & & & \vdots & & & & & \vdots & \\ a_{n,1} & \cdots & a_{n,n_z} & b_{n,1} & \cdots & b_{n,n_z} & c_{n,1} & \cdots & c_{n,n_z} & d_{1,1} & \cdots & d_{n,n_z} & e_n \end{bmatrix}$$

```
C sumDeltaInv4thOut()
{
    C sum(0);
    for(int j = -RJ; j <= RJ; ++j)
    {
        C a = -C(RK) - C(1)/C(2) + C(j) * tau;
        C b = C(RK) + C(1)/C(2) + C(j) * tau;

        sum += -C(1) / (C(3) * pow(a, 3))
            + C(1) / (C(6) * pow(a, 5))
            + C(1) / (C(3) * pow(b, 3))
            - C(1) / (C(6) * pow(b, 5));
    }
    sum += errorTerm(C(52) / (C(10) * pow(C(RK), 6)));
    return sum;
}

C sumDeltaInv6thOut()
{
    C sum(0);
    for(int j = -RJ; j <= RJ; ++j)
    {
        C a = (-C(RK) - C(1)/C(2) + C(j) * tau);
        C b = (C(RK) + C(1)/C(2) + C(j) * tau);

        sum += -C(1) / (C(5) * pow(a, 5))
            + C(1) / (C(5) * pow(b, 5));
    }
    sum += errorTerm(C(42) / (C(10) * pow(C(RK), 6)));
    return sum;
}
```

Figure 4.12 : Source for `sumDeltaInv4thOut` and `sumDeltaInv6thOut` corresponding to Proposition 4.3.2

where

$$a_{j,i} = Re\text{Res}|_{e_i^-}\frac{\Phi_j}{dz}$$
$$b_{j,i} = Re\text{Res}|_{e_i^+}\frac{\Phi_j}{dz}$$
$$c_{j,i} = Re\text{Res}|_{z_i^-}\frac{\Phi_j}{dz}$$
$$d_{j,i} = Im\text{Res}|_{z_i^-}\frac{\Phi_j}{dz}$$
$$e_j = Im\int_A \frac{\Phi_j}{dz}$$

Proof. Since the Φ_j are each antisymmetric, and dz is antisymmetric, it follows that the $\frac{\Phi_j}{dz}$ are symmetric. Then, by Proposition 3.1.2, the imaginary parts of their residues are 0, and the real part of their A-period is 0. The residue of $\frac{\Phi_j}{dz}$ at z_i^+ is the conjugate of the corresponding residue at z_i^-, so the constants $a_{j,i}$, $b_{j,i}$, $c_{j,i}$, $d_{j,i}$, and e_j completely determine $\frac{\Phi_j}{dz}$. If some linear combination of the rows gives 0, then some linear combination of the quadratic differentials gives 0, and vice versa. □

We note that this matrix has strictly more columns than rows, and at least one of the columns can be removed without changing the above result, since the sum of the residues of one-form are known to be zero. In the case of the Karcher surface we will analyze, we will also show that the final column of this matrix is not required to ensure independence. In other words, we will show that the following matrix is full rank:

$$\begin{bmatrix} a_{1,1} & \dots & a_{1,n_z} & b_{1,1} & \dots & b_{1,n_z} & c_{1,1} & \dots & c_{1,n_z} & d_{1,1} & \dots & d_{1,n_z} \\ \vdots & & & & & & \vdots & & & & & \vdots \\ a_{n,1} & \dots & a_{n,n_z} & b_{n,1} & \dots & b_{n,n_z} & c_{n,1} & \dots & c_{n,n_z} & d_{1,1} & \dots & d_{n,n_z} \end{bmatrix}$$

```
int cEN(int i) { return i; }
int cEP(int i) { return i + N; }
int cZR(int i) { return i + 2*N; }
int cZI(int i) { return i + 3*N; }
const int NCOLS = 4*N;

const int rDG = 0;
int rDH(int i) { return 1 + i; }
int rDI(int i) { return 1 + N + i; }
const int rDJ = 1 + 2*N;
const int rDK = 2 + 2*N;
const int NROWS = 3 + 2*N;

R gMatrix[NROWS][NCOLS];
```

Figure 4.13 : Functions and constants specifying the ordering of rows and columns in our matrix.

We implement a simple set of procedures for manipulating our matrix. In our program, we assign row and column indices to our matrix in the manner described above. In the program, the functions cEN, cEP, cZR, cZI produce column indices for $Re * |e^-$, $Re * |e^+$, $Re * |z^-$, and $Im * |z^-$ respectively. The rows are similarly named after the function whose differential they correspond to. For example, rDG is the index of the row corresponding the pole structure of $i\psi P_B$, the differential of the constraint function G. Similarly for rDH, rDI, rDJ, and rDK. See Figure 4.13.

4.5 Basic analysis of the domain

Our first task is to determine the geometry and parameters defining the Karcher surface. We recall that Karcher's surface is topologically a torus, and as it admits a reflective symmetry it is either a rectangular or rhombic torus. As its symmetry of reflection contains two disconnected fixpoint sets, it must be a rectangular torus.

One fixpoint set contains all of the ends.

The surface also admits a n_z-fold rotational symmetry ρ that satisfies:

$$\rho^* \eta = \rho$$

$$\rho^* g = \xi g$$

where $\xi^{n_z} = 1$.

Furthermore, the rotational symmetry is generated by the reflective symmetries $\sigma_1, ..., \sigma_{n_z}$. In practice, computations have been performed for $n_z = 3$ on the torus with a modulus of $\tau = 3i$. We choose to put 0 on the fixpoint locus of σ containing the ends and also on the fixpoint locus of one of the other reflective symmetries. See Figure 4.14.

We introduce real parameters L and d that determine the locations of the punctures as follows (letting $x(p)$ denote the location of puncture p):

$$x(e_k^-) = (k - \frac{n_z + 1}{2})\frac{\tau}{n_z} + d$$
$$x(e_k^+) = (k - \frac{n_z + 1}{2})\frac{\tau}{n_z} - d$$
$$x(z_k^-) = (k - \frac{n_z + 1}{2})\frac{\tau}{n_z} - L$$
$$x(z_k^+) = (k - \frac{n_z + 1}{2})\frac{\tau}{n_z} + L$$

We apply Abel's theorem applies to the divisor of g, which has poles at the z_k^- and zeroes at the z_k^+, and find that $2n_z L \in Z$. We choose $L = \frac{1}{n_z}$. These formulas justify the code definitions of Figure 4.15. We recall here our zero-based array subscripting convention - e_i^- in the program corresponds to `EN(i-1)` in the program.

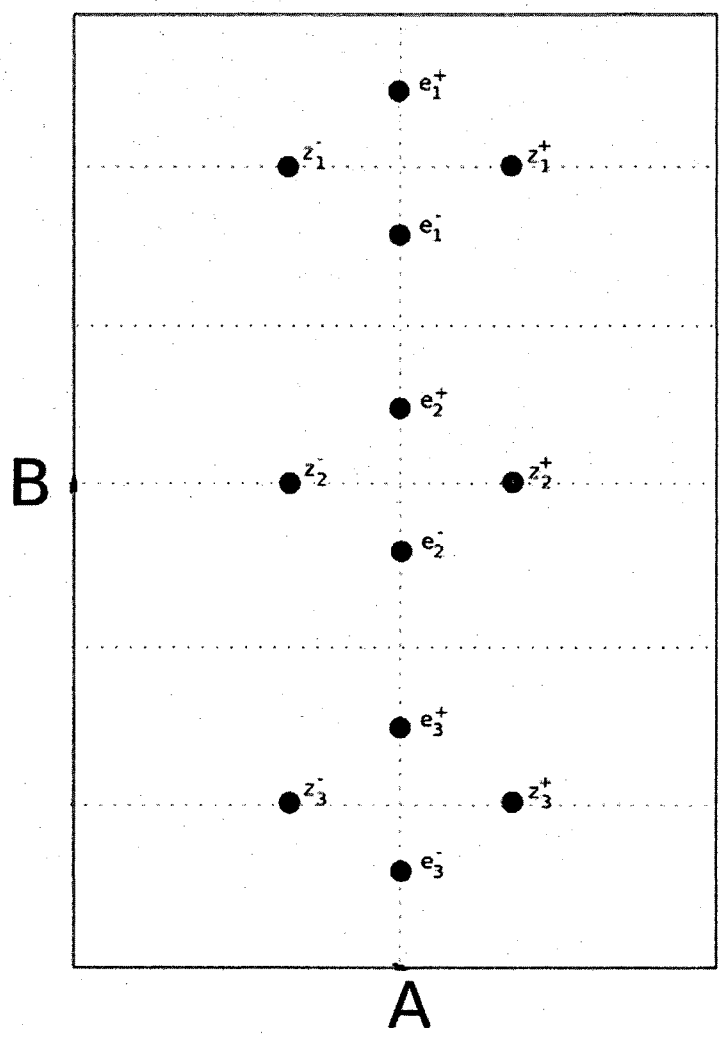

Figure 4.14 : Diagram of the punctured torus underlying Karcher's surface

```
const C tau(0.0, 3.0);
const int N = 3;

const C L = C(1)/C(6);
const C rootN = exp(PI * C(0,-2)/C(N));

C EN(int i)
{
    return tau * C(2*i - N + 1) / C(2*N) - d;
}

C EP(int i)
{
    return tau * C(2*i - N + 1) / C(2*N) + d;
}

C ZP(int i)
{
    return tau * C(2*i - N + 1) / C(2*N) - L;
}

C ZN(int i)
{
    return tau * C(2*i - N + 1) / C(2*N) + L;
}
```

Figure 4.15 : Source code of the functions implementing the layout of the geometry of the Riemann surface underlying our Karcher's surface.

4.6 Solving the constraints for the base surface

By choosing L to satisfy Abel's relation for g, we know that $\mathrm{Im}\int_B \psi$ is a multiple of 2π. The existence of the symmetries also ensures the satisfaction of other constraints immediately. Noting that $\rho(B) = B$, we have,

$$\int_B g\eta = \int_{\rho^{-1}(B)} \rho^*(g\eta) = \xi \int_B g\eta = 0$$
$$\int_B \frac{1}{g}\eta = \int_{\rho^{-1}(B)} \rho^*(\frac{1}{g}\eta) = \bar{\xi} \int_B \frac{1}{g}\eta = 0$$

Thus, the period problem is solved. Because η has poles of imaginary residue that lie along B, a curve with $\sigma_1(B) = -B$, and because η has imaginary B-period, we have that η is σ_1-symmetric, meaning $\overline{\sigma_1^*(\eta)} = \eta$. It follows that the value of η at a fixpoint of σ_1 must be purely real. This argument shows that $\mathrm{Im}\eta(z_1^+) = \mathrm{Im}\eta(z_1^-) = 0$. Furthermore, the symmetries σ and ρ ensure that if $\eta(z_1^+) = 0$, then $\eta(z_i^+) = 0$ and $\eta(z_i^-) = 0$ for each i.

Thus we have only a single real-valued constraint to solve: $\mathrm{Re}\frac{\eta}{dz} = 0$. Our first numerical task will be to find parameters such that this requirement is satisfied.

Our approach is to fix τ once and for all, and employ a binary search for upper and lower bounds on d. For each value of d under consideration, we can compute an interval bound on the value of $\frac{\eta}{dz}(z_1^-)$ on the surface Σ with that value of d. The size of the interval which we obtain can be made arbitrarily tight by increasing the number of terms included in the sum, which increases the computational effort required. We begin with two seed values for d, one ϵ and the other $\mathrm{Im}\frac{\tau}{2n_z} - \epsilon$, and evaluate η at both points. We find that the former is bounded above zero and the latter below zero. Using a binary search, we obtain find values d_U and d_L, with $d_U - d_L$ sufficiently small,

such that,

$$\frac{\eta}{dz}|_{d=d_L}(z_1^-) \geq 0 \tag{4.13}$$

$$\frac{\eta}{dz}|_{d=d_U}(z_i^-) \leq 0 \tag{4.14}$$

By the intermediate value thereom, a suitable value d lies in the interval $[d_L, d_U]$.

This process was carried out for the Karcher surface with $\tau = 3i$ and the values of d_L and d_U were recorded as program constants `D_LOWER` and `D_UPPER`, respectively. The first thing our program verifies is that with $d = d_L$, (4.13) holds and that with $d = d_U$, (4.14) holds. The program then continues to verify the quadratic differentials independent with d set to the interval $[d_L, d_U]$. The implementation of the approach described is shown in Figure 4.16.

4.7 Computing the residue structure of $i\psi P_B$

Our first quadratic differential is the most simple to compute. We know the residues and A-period of ψ from its definition.

For w centered at z_i^-, we have locally,

$$\frac{i\psi P_B}{dz} = \frac{-i}{w}dw + \ldots$$

For w centered at z_i^+, we have locally,

$$\frac{i\psi P_B}{dz} = \frac{i}{w}dw + \ldots$$

This description of the residue structure proves,

```
const FP D_LOWER = 0.11099364538241;
const FP D_UPPER = 0.11099364538656;

void verifyD()
{
    d = C(0, D_LOWER);
    FP a = evaluateEta(ZP(0)).real().lower();

    d = C(0, D_UPPER);
    FP b = evaluateEta(ZP(0)).real().upper();

    cout << "Lower bound on eta(ZP(0)) for d = D_LOWER is "
        << a << endl;
    cout << "Upper bound on eta(ZP(0)) for d = D_UPPER is "
        << b << endl;

    if(a < 0 || b > 0)
        throw std::runtime_error("Failed to verify d");

    d = C(0, R(D_LOWER, D_UPPER));

    cout << "Verified choice of "
            "d = [D_LOWER, D_UPPER]" << endl;
}
```

Figure 4.16 : Source for `verifyD` that verifies and sets a suitable value d

```
void computeDGResidues()
{
    cout << "Computing residues for dG" << endl;
    for(int k = 0; k < N; ++k)
    {
        gMatrix[rDG][cZI(k)] = -1;
    }
}
```

Figure 4.17 : Source for `computeDGResidues`

Proposition 4.7.1. *The residues of* $\frac{i\psi P_B}{dz}$ *are given by,*

$$\begin{aligned}
\mathrm{Res}|_{e_i^-} \frac{i\psi P_B}{dz} &= 0 \\
\mathrm{Res}|_{e_i^+} \frac{i\psi P_B}{dz} &= 0 \\
\mathrm{Res}|_{z_i^-} \frac{i\psi P_B}{dz} &= -i
\end{aligned}$$

This proposition is translated to the implementation of `computeDGResidues` shown in Figure 4.17.

4.8 Computing the residue structure of $Q_j^+\eta$ and $iQ_j^-\eta$

We recall the definition of Q_j:

1. Q_j is regular except at z_j^+ where it has no residue and satisfies $Q_j = d\frac{\alpha}{R}$ for some locally meromorphic one form α with residue 1.

2. $\int_{A_i} Q_j = 0$

Since $R = dz$, and α is given locally at z_j^+ by $\frac{dw}{w} + ...$, we have that Q_j is given locally, with w a coordinate centered at z_i^+, by

$$\frac{-dw}{w^2} + \frac{0dw}{w} + ...$$

For w centered at z_i^-, we have locally,

$$Q_j^- = -\frac{1}{2}\frac{dw}{w^2} + \frac{0dw}{w} + ...$$

To see that this is correct, consider $\frac{-\psi^2}{dz}$, which is also antisymmetric and has the same pole structure around z_j^+ and z_j^-.

For w centered at z_i^-, we have locally,

$$Q_j^+ = \frac{1}{2}\frac{dw}{w^2} + \frac{0dw}{w} + ...$$

Both Q_j^+ and Q_j^- have vanishing A-periods. Thus we have enough information to compute values of each of $\frac{Q_j^+}{dz}$ and $\frac{Q_j^-}{dz}$ at $e_i^\pm$.

Now we can put together the pole structure of the first quadratic differential, in local coordinates. For w centered at z_i^-, we have locally,

$$\begin{aligned} Q_j^+\eta &= (-\frac{1}{2}\frac{dw}{w^2} + \frac{0dw}{w} + ...)(0 + \frac{\partial}{\partial z}|_{z_i^-}(\frac{\eta}{dz})wdw + ...) \\ &= -\frac{1}{2}\frac{\partial}{\partial z}|_{z_i^-}(\frac{\eta}{dz})\frac{dw^2}{w} + ... \end{aligned}$$

For w centered at e_i^+, we have locally,

$$Q_j^+\eta = \frac{Q_j^+}{dz}(e_i^+)\frac{dw^2}{2\pi i w} + ...$$

For w centered at e_i^-, we have locally,

$$Q_j^+\eta = -\frac{Q_j^+}{dz}(e_i^-)\frac{dw^2}{2\pi i w} + \ldots$$

Similarly, for the second differential, for w centered at z_i^-, we have locally,

$$\begin{aligned} Q_j^-\eta &= (\frac{i}{2}\frac{dw}{w^2} + \frac{0dw}{w} + \ldots)(0 + \frac{\partial}{\partial z}|_{z_i^-}(\frac{\eta}{dz})wdw + \ldots) \\ &= \frac{i}{2}\frac{\partial}{\partial z}|_{z_i^-}(\frac{\eta}{dz})\frac{dw^2}{w} + \ldots \end{aligned}$$

For w centered at e_i^+, we have locally,

$$iQ_j^-\eta = \frac{Q_j^-}{dz}(e_i^+)\frac{dw^2}{2\pi w} + \ldots$$

For w centered at e_i^-, we have locally,

$$iQ_j^-\eta = -\frac{Q_j^-}{dz}(e_i^-)\frac{dw^2}{2\pi w} + \ldots$$

Proposition 4.8.1. *The residues of $Q_j^+\eta$ are given by,*

$$\begin{aligned} \text{Res}|_{z_i^-}Q_j^+\eta &= -\frac{1}{2}\frac{\partial}{\partial z}|_{z_i^-}(\frac{\eta}{dz}) \\ \text{Res}|_{e_i^+}Q_j^+\eta &= \frac{1}{2\pi i}\frac{Q_j^+}{dz}(e_i^+) \\ \text{Res}|_{e_i^-}Q_j^+\eta &= \frac{1}{2\pi i}\frac{Q_j^+}{dz}(e_i^-) \end{aligned}$$

The residues of $iQ_j^-\eta$ are given by,

$$\begin{aligned}
\mathrm{Res}|_{z_i^-} Q_j^-\eta &= \frac{i}{2}\frac{\partial}{\partial z}|_{z_i^-}(\frac{\eta}{dz}) \\
\mathrm{Res}|_{e_i^+} Q_j^-\eta &= \frac{1}{2\pi}\frac{Q_j^-}{dz}(e_i^+) \\
\mathrm{Res}|_{e_i^-} Q_j^-\eta &= \frac{1}{2\pi}\frac{Q_j^-}{dz}(e_i^-)
\end{aligned}$$

This proposition is translated to the implementations presented in figures 4.19 and 4.18. The implementation of `evaluateDEta` computes the value of $\frac{\partial}{\partial z}(\frac{\eta}{dz})$. As an exact form, this has zero A-period. At e_i^-, where η has a simple pole of residue $\frac{-1}{2\pi i}$, $\frac{\partial}{\partial z}(\frac{\eta}{dz})$ has a second order pole with coefficient $\frac{1}{2\pi i}$. Similarly, at e_i^+, where η has a simple pole of residue $\frac{1}{2\pi i}$, $\frac{\partial}{\partial z}(\frac{\eta}{dz})$ has a second order pole with coefficient $\frac{-1}{2\pi i}$.

4.9 Computing values of g and g^{-1}

Rather than constructing g from its definition in terms of ψ, we take an alternate approach that is more amenable to computation using our methods.

We note that gdz is a one-form with simple poles at the z_i^- and with residues multiplied by ξ as the surface is translated by ρ. This motivates us to specify a one-form g_0 with residue 1, arbitrarily, at z_1^-, and with $\mathrm{Res}|_{\rho(z_i^-)} g_0 = \xi \mathrm{Res}|_{z_i^-} g_0$. We recover g using,

$$g = \frac{g_0 - g_0(z_i^+)}{g_0(e_1^-) - g_0(z_i^+)}$$

This function has poles and zeroes at all the right places, as well as being normalized to 1 at e_0^-, as required.

Rather than using g to compute g^{-1}, we follow a similar process to compute g^{-1}

```
C evaluateQPJ(int j, C z)
{
    C sum(0);

    sum += -C(1)/C(2)
            * evaluateSecondOrderPoleZeroAPeriod(ZP(j),z)
        +  C(1)/C(2)
            * evaluateSecondOrderPoleZeroAPeriod(ZN(j),z);

    return sum;
}

C evaluateQNJ(int j, C z)
{
    C sum(0);

    sum += -C(1)/C(2)
            * evaluateSecondOrderPoleZeroAPeriod(ZP(j),z)
        - C(1)/C(2)
            * evaluateSecondOrderPoleZeroAPeriod(ZN(j),z);

    return sum;
}
```

Figure 4.18 : Source for evaluateQPJ and evaluateQNJ

```
C evaluateDEta(C z)
{
    C sum(0);
    for(int i=0; i<N; ++i)
    {
        sum += C(0, 1) / (C(2) * PI)
            * evaluateSecondOrderPoleZeroAPeriod(EN(i), z)
            + C(0, -1) / (C(2) * PI)
            * evaluateSecondOrderPoleZeroAPeriod(EP(i), z);
    }
    return sum;
}

void computeDHResidues()
{
    for(int j = 0; j < N; ++j)
    {
        cout << "Computing residues for dH_" << j << endl;
        for(int k = 0; k < N; ++k)
        {
            C resZN = -C(1)/C(2) * evaluateDEta(ZN(k));
            gMatrix[rDH(j)][cZR(k)] = resZN.real();
            gMatrix[rDH(j)][cZI(k)] = resZN.imag();

            C resEP = C(1)/(C(0,2)*PI) * evaluateQPJ(j, EP(k));
            gMatrix[rDH(j)][cEP(k)] = resEP.real();

            C resEN = C(1)/(C(0,2)*PI) * evaluateQPJ(j, EN(k));
            gMatrix[rDH(j)][cEN(k)] = resEN.real();
        }
    }
}
```

Figure 4.19 : Source for evaluateDEta and computeDHResidues

directly. This motivates us to specify a one-form g_0^{-1} with residue 1, arbitrarily, at z_1^+, and with $\text{Res}|_{\rho(z_i^+)} g_0^{-1} = \xi \text{Res}|_{z_i^+} g_0^{-1}$. We recover g^{-1} using,

$$g^{-1} = \frac{g_0^{-1} - g_0^{-1}(z_i^-)}{g_0^{-1}(e_1^-) - g_0^{-1}(z_i^-)}$$

This function has poles and zeroes at all the right places, as well as being normalized to 1 at e_0^-, as required.

Thus we have obtained representations of g and g^{-1} that can be used for effectively computing values of these functions. See Figure 4.20 for the corresponding implementation.

4.10 Computing S and T

We will have need to compute values of the helper forms $\frac{S}{dz}$ and $\frac{T}{dz}$ that appear in the last two quadratic differentials. First we construct S. We choose S' to be a function with second order pole with no residue and z^{-2} coefficient of 1 at z_1^+, and an A-period of 0. We then look for a linear combination of S' and dz to make the periods vanish. The A-period of dz is 1, and the B-period is τ. The A-period of S' is 0, and the B-period is something we can compute. We thus solve for the desired combination:

$$S = dz - \frac{\tau S'}{\int_B S'}$$

We construct T similarly. We choose T' to be a function with second order pole with no residue and z^{-2} coefficient of 1 at z_1^-, and an A-period of 0. We then look for a linear combination of T' and dz to make the periods vanish. The A-period of dz is 1, and the B-period is τ. The A-period of T' is 0, and the B-period is something

```
C evaluateG0(C z)
{
    C sum(0);
    C coeff(1);
    for(int i=0; i<N; ++i)
    {
        sum += coeff * evaluateSimplePole(ZN(i), z);
        coeff *= rootN;
    }
    return sum;
}

C evaluateG(C z)
{
    return (evaluateG0(z) - evaluateG0(ZP(0)))
        / (evaluateG0(EN(0)) - evaluateG0(ZP(0)));
}

C evaluateGInv0(C z)
{
    C sum(0);
    C coeff(1);
    for(int i=0; i<N; ++i)
    {
        sum += coeff * evaluateSimplePole(ZP(i), z);
        coeff /= rootN;
    }
    return sum;
}

C evaluateGInv(C z)
{
    return (evaluateGInv0(z) - evaluateGInv0(ZN(0)))
        / (evaluateGInv0(EN(0)) - evaluateGInv0(ZN(0)));
}
```

Figure 4.20 : Source for computing g and g^{-1}

we can compute. We thus solve for the desired combination:

$$T = dz - \frac{\tau T'}{\int_B T'}$$

Thus we have obtained representations of S and T that can be used for effectively computing values of $\frac{S}{dz}$ and $\frac{T}{dz}$. These discussions motiviate the implementations shown in figures 4.21 and 4.21.

4.11 Computing the residue structure of the remaining differentials

We now describe the process for computing the differentials given by,

- $\Phi_1 = \frac{1}{2}(T\frac{1}{g} - Sg)\eta + \sum_i \left[c_1(e_i^+)P_{e_i^+} - c_1(e_i^-)P_{e_i^-}\right]\psi$
- $\Phi_2 = \frac{i}{2}(T\frac{1}{g} + Sg)\eta - i\sum_i \left[c_2(e_i^+)P_{e_i^+} + c_2(e_i^-)P_{e_i^-}\right]\psi$

where

$$c_1(p) = \mathrm{Re}\left(\int_{C_p} S\right) g(p)$$
$$c_2(p) = \mathrm{Im}\left(\int_{C_p} S\right) g(p)$$

We first note that $(T\frac{1}{g} - Sg)\eta$ is regular at $z_i^{\pm}$, as the zeroes of η and $g^{\pm 1}$ cancel the second order poles of T and S. For w centered at e_i^+, we have locally,

$$(T\frac{1}{g} - Sg)\eta = \left[\frac{T_j}{dz}(e_i^+)\frac{1}{g(e_i^+)} - \frac{S_j}{dz}(e_i^+)g(e_i^+)\right]\frac{dw}{2\pi i w} + \ldots$$

```
C evaluateSPrime(C z)
{
    return evaluateSecondOrderPoleZeroAPeriod(ZP(0), z);
}

C integrateSPrimeAlongB()
{
    return integrateSecondOrderPoleZeroAPeriodAlongB(ZP(0));
}

C integrateSPrime(C a, C b)
{
    return integrateSecondOrderPoleZeroAPeriod(ZP(0), a, b);
}

C evaluateS(C z)
{
    return C(1) - tau * evaluateSPrime(z)
            / integrateSPrimeAlongB();
}

C integrateS(C a, C b)
{
    return (b - a) - tau * integrateSPrime(a, b)
            / integrateSPrimeAlongB();
}
```

Figure 4.21 : Source for evaluating and integrating S

```
C evaluateTPrime(C z)
{
    return evaluateSecondOrderPoleZeroAPeriod(ZN(0), z);
}

C integrateTPrime(C a, C b)
{
    return integrateSecondOrderPoleZeroAPeriod(ZN(0), a, b);
}

C integrateTPrimeAlongB()
{
    return integrateSecondOrderPoleZeroAPeriodAlongB(ZN(0));
}

C evaluateT(C z)
{
    return C(1) - tau * evaluateTPrime(z)
           / integrateTPrimeAlongB();
}

C integrateT(C a, C b)
{
    return (b - a) - tau * integrateTPrime(a, b)
           / integrateTPrimeAlongB();
}
```

Figure 4.22 : Source for evaluating and integrating T

For w centered at e_i^-, we have locally,

$$(T\frac{1}{g} - Sg)\eta = -\left[\frac{T}{dz}(e_i^-)\frac{1}{g(e_i^-)} - \frac{S}{dz}(e_i^-)g(e_i^-)\right]\frac{dw}{2\pi i w} + ...$$

Similarly, for w centered at e_i^+, we have locally,

$$i(T\frac{1}{g} + Sg)\eta = \left[\frac{T}{dz}(e_i^+)\frac{1}{g(e_i^+)} + \frac{S}{dz}(e_i^+)g(e_i^+)\right]\frac{dw}{2\pi w} + ...$$

For w centered at e_i^-, we have locally,

$$i(T\frac{1}{g} + Sg)\eta = -\left[\frac{T}{dz}(e_i^-)\frac{1}{g(e_i^-)} + \frac{S}{dz}(e_i^-)g(e_i^-)\right]\frac{dw}{2\pi w} + ...$$

Next, we compute the residues of the second terms of Φ_1,

$$\Phi_3 = \sum_i \left[c_1(e_i^+)P_{e_i^+} - c_1(e_i^-)P_{e_i^-}\right]\psi$$

For w centered at e_k^+, we have locally,

$$\Phi_3 = c_1(e_k^+)\frac{\psi}{dz}(e_k^+)\frac{dw}{w} + ...$$

For w centered at e_k^-, $k \neq 1$, we have locally,

$$\Phi_3 = -c_1(e_k^-)\frac{\psi}{dz}(e_k^-)\frac{dw}{w} + ...$$

For w centered at e_1^-, we have locally,

$$\Phi_3 = -\sum_i \left[c_1(e_i^+) - c_1(e_i^-)\right]\frac{\psi}{dz}(e_1^-)\frac{dw}{w} + ...$$

For w centered at z_k^-, we have locally,

$$\Phi_3 = \sum_i \left[c_1(e_i^+) \frac{P_{e_i^+}}{dz}(z_k^-) - c_1(e_i^-) \frac{P_{e_i^-}}{dz}(z_k^-) \right] \frac{dw}{w} + \ldots$$

And we compute the residues of the second terms of Φ_2,

$$\Phi_4 = i \sum_i \left[c_2(e_i^+) P_{e_i^+} + c_2(e_i^-) P_{e_i^-} \right] \psi$$

For w centered at e_k^+, we have locally,

$$\Phi_4 = i c_2(e_k^+) \frac{\psi}{dz}(e_k^+) \frac{dw}{w} + \ldots$$

For w centered at e_k^-, $k \neq 1$, we have locally,

$$\Phi_4 = i c_2(e_k^-) \frac{\psi}{dz}(e_k^-) \frac{dw}{w} + \ldots$$

For w centered at e_1^-, we have locally,

$$\Phi_4 = i \sum_i \left[c_2(e_i^+) + c_2(e_i^-) \right] \frac{\psi}{dz}(e_1^-) \frac{dw}{w} + \ldots$$

For w centered at z_k^-, we have locally,

$$\Phi_4 = -i \sum_i \left[c_2(e_i^+) \frac{P_{e_i^+}}{dz}(z_k^-) + c_2(e_i^-) \frac{P_{e_i^-}}{dz}(z_k^-) \right] \frac{dw}{w} + \ldots$$

Putting it all together, we obtain the pole structure for Φ_1. For w centered at e_k^+, we have locally,

$$\Phi_1 = \left(\frac{1}{2\pi i} \left[\frac{T_j}{dz}(e_k^+) \frac{1}{g(e_k^+)} - \frac{S_j}{dz}(e_k^+) g(e_k^+) \right] + c_1(e_k^+) \frac{\psi}{dz}(e_k^+) \right) \frac{dw}{w} + \ldots$$

For w centered at e_k^-, $k \neq 1$, we have locally,

$$\Phi_1 = \left(\frac{-1}{2\pi i} \left[\frac{T}{dz}(e_k^-) \frac{1}{g(e_k^-)} - \frac{S}{dz}(e_k^-) g(e_k^-) \right] - c_1(e_k^-) \frac{\psi}{dz}(e_k^-) \right) \frac{dw}{w} + \ldots$$

For w centered at e_1^-, we have locally,

$$\Phi_1 = \left(\frac{-1}{2\pi i} \left[\frac{T}{dz}(e_1^-) \frac{1}{g(e_1^-)} - \frac{S}{dz}(e_1^-) g(e_1^-) \right] + \sum_i \left[c_1(e_i^+) - c_1(e_i^-) \right] \frac{\psi}{dz}(e_1^-) \right) \frac{dw}{w} + \ldots$$

For w centered at z_k^-, we have locally,

$$\Phi_1 = \sum_i \left[c_1(e_i^+) \frac{P_{e_i^+}}{dz}(z_k^-) - c_1(e_i^-) \frac{P_{e_i^-}}{dz}(z_k^-) \right] \frac{dw}{w} + \ldots$$

Finally, we obtain the pole structure for Φ_2. For w centered at e_k^+, we have locally,

$$\Phi_2 = \left(\frac{1}{2\pi} \left[\frac{T}{dz}(e_k^+) \frac{1}{g(e_k^+)} + \frac{S}{dz}(e_k^+) g(e_k^+) \right] + ic_2(e_k^+) \frac{\psi}{dz}(e_k^+) \right) \frac{dw}{w} + \ldots$$

For w centered at e_k^-, $k \neq 1$, we have locally,

$$\Phi_2 = \left(\frac{-1}{2\pi} \left[\frac{T}{dz}(e_k^-) \frac{1}{g(e_k^-)} + \frac{S}{dz}(e_k^-) g(e_k^-) \right] + ic_2(e_k^-) \frac{\psi}{dz}(e_k^-) \right) \frac{dw}{w} + \ldots$$

For w centered at e_1^-, we have locally,

$$\Phi_2 = \left(\frac{-1}{2\pi} \left[\frac{T}{dz}(e_1^-) \frac{1}{g(e_1^-)} + \frac{S}{dz}(e_1^-) g(e_1^-) \right] - i \sum_i \left[c_2(e_i^+) + c_2(e_i^-) \right] \frac{\psi}{dz}(e_1^-) \right) \frac{dw}{w} + \ldots$$

For w centered at z_k^-, we have locally,

$$\Phi_2 = -i \sum_i \left[c_2(e_i^+) \frac{P_{e_i^+}}{dz}(z_k^-) + c_2(e_i^-) \frac{P_{e_i^-}}{dz}(z_k^-) \right] \frac{dw}{w} + \ldots$$

Proposition 4.11.1. *The residues of the differential $dJ = \Phi_1$ are given by,*

$$\begin{aligned}
\mathrm{Res}|_{e_k^+} \Phi_1 &= \frac{1}{2\pi i} \left[\frac{T}{dz}(e_k^+) \frac{1}{g(e_k^+)} - \frac{S}{dz}(e_k^+) g(e_k^+) \right] + c_1(e_k^+) \frac{\psi}{dz}(e_k^+) \\
\mathrm{Res}|_{e_k^-} \Phi_1 &= \frac{-1}{2\pi i} \left[\frac{T}{dz}(e_k^-) \frac{1}{g(e_k^-)} - \frac{S}{dz}(e_k^-) g(e_k^-) \right] - c_1(e_k^-) \frac{\psi}{dz}(e_k^-) \qquad k \neq 1 \\
\mathrm{Res}|_{e_1^-} \Phi_1 &= \frac{-1}{2\pi i} \left[\frac{T}{dz}(e_1^-) \frac{1}{g(e_1^-)} - \frac{S}{dz}(e_1^-) g(e_1^-) \right] + \sum_i \left[c_1(e_i^+) - c_1(e_i^-) \right] \frac{\psi}{dz}(e_1^-) \\
\mathrm{Res}|_{z_k^-} \Phi_1 &= \sum_i \left[c_1(e_i^+) \frac{P_{e_i^+}}{dz}(z_k^-) - c_1(e_i^-) \frac{P_{e_i^-}}{dz}(z_k^-) \right]
\end{aligned}$$

The residues of the differential $dK = \Phi_2$ are given by

$$\begin{aligned}
\mathrm{Res}|_{e_k^+} \Phi_2 &= \frac{1}{2\pi} \left[\frac{T}{dz}(e_k^+) \frac{1}{g(e_k^+)} + \frac{S}{dz}(e_k^+) g(e_k^+) \right] + i c_2(e_k^+) \frac{\psi}{dz}(e_k^+) \\
\mathrm{Res}|_{e_k^-} \Phi_2 &= \frac{-1}{2\pi} \left[\frac{T}{dz}(e_k^-) \frac{1}{g(e_k^-)} + \frac{S}{dz}(e_k^-) g(e_k^-) \right] + i c_2(e_k^-) \frac{\psi}{dz}(e_k^-) \qquad k \neq 1 \\
\mathrm{Res}|_{e_1^-} \Phi_2 &= \frac{-1}{2\pi} \left[\frac{T}{dz}(e_1^-) \frac{1}{g(e_1^-)} + \frac{S}{dz}(e_1^-) g(e_1^-) \right] - i \sum_i \left[c_2(e_i^+) + c_2(e_i^-) \right] \frac{\psi}{dz}(e_1^-) \\
\mathrm{Res}|_{z_k^-} \Phi_2 &= -i \sum_i \left[c_2(e_i^+) \frac{P_{e_i^+}}{dz}(z_k^-) + c_2(e_i^-) \frac{P_{e_i^-}}{dz}(z_k^-) \right]
\end{aligned}$$

```
C evaluatePsi(C z)
{
    C sum(0);
    for(int i = 0; i < N; ++i)
    {
        sum += evaluateSimplePole(ZP(i), z)
            - evaluateSimplePole(ZN(i), z);
    }
    return sum;
}

C evaluateC1(C p)
{
    return ( integrateS(-C(1)/C(2) - tau / C(2), p)
            * evaluateG(p)
        ).real();
}

C evaluateC2(C p)
{
    return ( integrateS(-C(1)/C(2) - tau / C(2), p)
            * evaluateG(p)
        ).imag();
}
```

Figure 4.23 : Source for supporting functions needed by dJ and dK

This proposition is translated to the implementation shown in figures 4.23, 4.24, and 4.25.

4.12 Main Theorem

Finally, once we've built our 12x9 matrix, we need to calculate whether it is full rank. Our approach is a Gaussian Elimination with full pivoting, implemented by source shown in figures 4.27 and 4.28. Our Gaussian Elimination routine applies a

```
void computeDJResidues()
{
    cout << "Computing residues for dJ" << endl;
    for(int k = 0; k < N; ++k)
    {
        C resZN(0);
        for(int i = 0; i < N; ++i)
        {
            resZN += evaluateC1(EP(i))
                    * evaluatePQ(EP(i), ZN(k))
                -  evaluateC1(EN(i))
                    * evaluatePQ(EN(i), ZN(k));
        }
        gMatrix[rDJ][cZR(k)] = resZN.real();
        gMatrix[rDJ][cZI(k)] = resZN.imag();

        C resEP = C(1)/(C(0,2)*PI)
                * ( evaluateT(EP(k)) * evaluateGInv(EP(k))
                    - evaluateS(EP(k)) * evaluateG(EP(k)) )
                + evaluateC1(EP(k)) * evaluatePsi(EP(k));

        gMatrix[rDJ][cEP(k)] = resEP.real();

        C resEN = C(-1)/(C(0,2)*PI)
                * ( evaluateT(EN(k)) * evaluateGInv(EN(k))
                    - evaluateS(EN(k)) * evaluateG(EN(k)) );

        if(k == 0)
            for(int i = 0; i < N; ++i)
                resEN += (evaluateC1(EP(i)) - evaluateC1(EN(i)))
                        * evaluatePsi(EN(0));
        else
            resEN -= evaluateC1(EN(k)) * evaluatePsi(EN(k));

        gMatrix[rDJ][cEN(k)] = resEN.real();
    }
}
```

Figure 4.24 : Source to construct the row of matrix for the differential dJ

```
void computeDKResidues()
{
    cout << "Computing residues for dK" << endl;
    for(int k = 0; k < N; ++k)
    {
        C resZN(0);
        for(int i = 0; i < N; ++i)
        {
            resZN += C(0,-1) * (
                        evaluateC2(EP(i))
                        * evaluatePQ(EP(i), ZN(k))
                        +  evaluateC2(EN(i))
                        * evaluatePQ(EN(i), ZN(k))
                    );
        }
        gMatrix[rDK][cZR(k)] = resZN.real();
        gMatrix[rDK][cZI(k)] = resZN.imag();

        C resEP = C(1)/(C(2)*PI)
                * ( evaluateT(EP(k)) * evaluateGInv(EP(k))
                + evaluateS(EP(k)) * evaluateG(EP(k)) )
                + C(0,1) * evaluateC2(EP(k)) * evaluatePsi(EP(k));

        gMatrix[rDK][cEP(k)] = resEP.real();

        C resEN = C(-1)/(C(2)*PI)
                * ( evaluateT(EN(k)) * evaluateGInv(EN(k))
                + evaluateS(EN(k)) * evaluateG(EN(k)) );

        if(k == 0)
            for(int i = 0; i < N; ++i)
                resEN -= C(0,1) * (evaluateC2(EP(i))
                        + evaluateC2(EN(i))) * evaluatePsi(EN(0));
        else
            resEN += C(0,1) * evaluateC2(EN(k)) * evaluatePsi(EN(k));

        gMatrix[rDK][cEN(k)] = resEN.real();
    }
}
```

Figure 4.25 : Source to construct the row of matrix for the differential dK

```
void printMatrix()
{
    cout << fixed;
    for(int r = 0; r < NROWS; ++r)
    {
        for(int c = 0; c < NCOLS; ++c)
        {
            cout.precision(2);
            cout.fill(' ');
            cout.width(6);
            cout << gMatrix[r][c].upper();
        }

        cout << endl;
    }
    cout << endl;
}

void printErrorMatrix()
{
    cout << fixed;
    for(int r = 0; r < NROWS; ++r)
    {
        for(int c = 0; c < NCOLS; ++c)
        {
            cout.precision(2);
            cout.fill(' ');
            cout.width(6);

            FP err = gMatrix[r][c].upper()
                - gMatrix[r][c].lower();

            cout << -log(err)/log(10);
        }

        cout << endl;
    }
    cout << endl;
}
```

Figure 4.26 : Source to matrix printing subroutines

```
void swap(R &a, R &b)
{
    R temp = a;
    a = b;
    b = temp;
}

void swapRows(int r1, int r2)
{
    cout << "Swapping rows " << r1 << " and " << r2 << endl;
    for(int c = 0; c < NCOLS; ++c)
        swap(gMatrix[r1][c], gMatrix[r2][c]);
}

void swapCols(int c1, int c2)
{
    cout << "Swapping columns " << c1 << " and " << c2 << endl;
    for(int r = 0; r < NROWS; ++r)
        swap(gMatrix[r][c1], gMatrix[r][c2]);
}
```

Figure 4.27 : Row and column swap subroutines, used by `rowReduceMatrix`

```
void rowReduceMatrix()
{
    cout << "Original matrix:" << endl;
    printMatrix();
    for(int n = 0; n < NROWS; ++n) {
        FP pivot(0.0);
        int pivotR = n, pivotC = n;
        for(int r = n; r < NROWS; ++r)
            for(int c = n; c < NCOLS; ++c) {
                FP x = max( gMatrix[r][c].lower(),
                            -gMatrix[r][c].upper());
                if(x > pivot) {
                    pivotR = r;
                    pivotC = c;
                    pivot = x;
                }
            }
        swapRows(n, pivotR);
        swapCols(n, pivotC);
        if(gMatrix[n][n].lower() <= 0.0 &&
           gMatrix[n][n].upper() >= 0.0) {
            cout << "Matrix inversion failure" << endl;
            return;
        }
        for(int r = n+1; r < NROWS; ++r) {
            R scale = gMatrix[r][n] / gMatrix[n][n];
            gMatrix[r][n] = R(0);
            for(int c = n + 1; c < NCOLS; ++c)
                gMatrix[r][c] -= gMatrix[n][c] * scale;
        }
        cout << "Matrix after elimination step:" << endl;
        printMatrix();
    }
    cout << "Log_10(1/error) in final matrix:" << endl;
    printErrorMatrix();
    cout << "Matrix verified full rank" << endl;
}
```

Figure 4.28 : Implementation of Gaussian elimination with pivoting

```
int main(int argc, char *argv[])
{
    verifyD();
    computeDGResidues();
    computeDHResidues();
    computeDIResidues();
    computeDJResidues();
    computeDKResidues();
    rowReduceMatrix();
    return 0;
}
```

Figure 4.29 : Source for the entry point of our program

number elementary operations (swapping two rows, swapping two columns, or adding a multiple of one row to another) that do no change the rank of the matrix, until it is in upper-triangular form with diagnal elements bounded away from zero. At each step of the computation, we have a matrix of intervals whose elements bound the corresponding elements in the matrix of reals whose rank we are trying to determine. Thus, if we can obtain this upper-triangular form without ever encountering pivot element (which becomes a diagnal of the reduced matrix) whose bounding interval contains 0 at any step of the process, we will have established the matrix to be full rank. As a final step, we print the number of significant digits present in each matrix entry, computed as the $\log_{10}$ of reciprocal interval sizes.

Our approach to interval matrix rank determination is somewhat simplistic. In the iterative Guassian Elimination passes, interval error can accumulate exponentially, and this accumulated interval error can make the resultant matrix look potentially rank deficient after some number of steps, even though the original interval matrix was necessarily full rank. More sophisticated methods are widely known. For example,

[Manteuffel81] compares two methods, one of which bounds the change in singular values of a matrix in terms of a change in the values of its entries, since if the singular values can be bounded away from zero then the matrx is full rank. The presented bound is linear in the number of rows of the matrix, rather than exponential, so for performing rank determination on larger interval matrices such an approach is essential. In our case, however, we estimate that we only lose between one and two significant digits by not employing such methods.

The `main` function is shown in Figure 4.29 that implements our top level plan. First, the interval pre-computed for d is verified to contain the d for Karcher's surface. Next, the matrix of residues is assembled for each type of quadratic differential. Finally, the assembled matrix is row reduced to verify that it is full rank. The output transcript for the program is shown in figures 4.30, 4.31, 4.32, and 4.33. Examining the output, we see that we have proven,

Main Theorem. *There exists a 3-dimensional family of complete embedded symmetric minimal surfaces in a neighborhood of the Karcher surface with 3-fold rotational symmetry and* $\tau = 3i$.

```
Lower bound on eta(ZP(0)) for d = D_LOWER is 3.94726e-12
Upper bound on eta(ZP(0)) for d = D_UPPER is -3.82009e-12
Verified choice of d = [D_LOWER, D_UPPER]
Computing residues for dG
Computing residues for dH_0
Computing residues for dH_1
Computing residues for dH_2
Computing residues for dI_0
Computing residues for dI_1
Computing residues for dI_2
Computing residues for dJ
Computing residues for dK
Original matrix:
  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00 -1.00 -1.00 -1.00
 -3.62  0.02 -0.01  3.62  0.01 -0.02 -3.61 -3.61 -3.61  0.00  0.00  0.00
 -0.01 -3.62  0.02 -0.02  3.62  0.01 -3.61 -3.61 -3.61  0.00  0.00  0.00
  0.02 -0.01 -3.62  0.01 -0.02  3.62 -3.61 -3.61 -3.61  0.00  0.00  0.00
 -1.71  0.37  0.36 -1.71  0.36  0.37  0.00  0.00  0.00  3.61  3.61  3.61
  0.36 -1.71  0.37  0.37 -1.71  0.36  0.00  0.00  0.00  3.61  3.61  3.61
  0.37  0.36 -1.71  0.36  0.37 -1.71  0.00  0.00  0.00  3.61  3.61  3.61
 -3.46 13.15 -0.80  9.45  5.27 -4.21 -2.17  0.27  0.75 -0.89  1.31 -0.43
  1.32  0.32  0.33 -2.28  0.66 -0.34  0.38 -0.60  0.23  2.53 -0.92 -0.40

Swapping rows 0 and 7
Swapping columns 0 and 1
Matrix after elimination step:
 13.15 -3.46 -0.80  9.45  5.27 -4.21 -2.17  0.27  0.75 -0.89  1.31 -0.43
  0.00 -3.62 -0.00  3.61 -0.00 -0.01 -3.60 -3.61 -3.61  0.00 -0.00  0.00
  0.00 -0.96 -0.20  2.58  5.08 -1.15 -4.21 -3.53 -3.40 -0.24  0.36 -0.12
  0.00  0.02 -3.62  0.01 -0.02  3.62 -3.61 -3.61 -3.61 -0.00  0.00 -0.00
  0.00 -1.61  0.38 -1.98  0.21  0.49  0.06 -0.01 -0.02  3.63  3.57  3.62
  0.00 -0.09  0.26  1.60 -1.02 -0.19 -0.28  0.04  0.10  3.49  3.78  3.55
  0.00  0.46 -1.69  0.10  0.22 -1.60  0.06 -0.01 -0.02  3.63  3.57  3.62
  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00 -1.00 -1.00 -1.00
  0.00  1.40  0.35 -2.51  0.53 -0.24  0.43 -0.61  0.21  2.55 -0.95 -0.39
```

Figure 4.30 : Output transcript of program execution (page 1)

```
Swapping rows 1 and 2
Swapping columns 1 and 4
Matrix after elimination step:
 13.15  5.27 -0.80  9.45 -3.46 -4.21 -2.17  0.27  0.75 -0.89  1.31 -0.43
  0.00  5.08 -0.20  2.58 -0.96 -1.15 -4.21 -3.53 -3.40 -0.24  0.36 -0.12
  0.00  0.00 -0.00  3.61 -3.62 -0.01 -3.61 -3.61 -3.61  0.00 -0.00  0.00
  0.00  0.00 -3.62  0.02  0.02  3.62 -3.62 -3.62 -3.62 -0.00  0.00 -0.00
  0.00  0.00  0.39 -2.08 -1.57  0.53  0.24  0.14  0.12  3.64  3.55  3.62
  0.00  0.00  0.22  2.12 -0.28 -0.42 -1.13 -0.68 -0.59  3.44  3.85  3.53
  0.00  0.00 -1.68 -0.01  0.51 -1.55  0.25  0.15  0.13  3.64  3.55  3.62
  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00 -1.00 -1.00 -1.00
  0.00  0.00  0.37 -2.78  1.50 -0.12  0.87 -0.24  0.57  2.58 -0.99 -0.38

Swapping rows 2 and 5
Swapping columns 2 and 10
Matrix after elimination step:
 13.15  5.27  1.31  9.45 -3.46 -4.21 -2.17  0.27  0.75 -0.89 -0.80 -0.43
  0.00  5.08  0.36  2.58 -0.96 -1.15 -4.21 -3.53 -3.40 -0.24 -0.20 -0.12
  0.00  0.00  3.85  2.12 -0.28 -0.42 -1.13 -0.68 -0.59  3.44  0.22  3.53
  0.00  0.00  0.00  0.02  0.02  3.62 -3.62 -3.62 -3.62 -0.00 -3.62 -0.00
  0.00  0.00  0.00 -4.04 -1.31  0.92  1.28  0.77  0.67  0.46  0.18  0.37
  0.00  0.00  0.00  3.61 -3.62 -0.01 -3.61 -3.61 -3.61  0.00 -0.00  0.00
  0.00  0.00  0.00 -1.97  0.77 -1.16  1.29  0.77  0.67  0.46 -1.89  0.37
  0.00  0.00  0.00  0.55 -0.07 -0.11 -0.29 -0.18 -0.15 -0.11  0.06 -0.08
  0.00  0.00  0.00 -2.24  1.43 -0.23  0.58 -0.41  0.42  3.46  0.42  0.52

Swapping rows 3 and 4
Swapping columns 3 and 3
Matrix after elimination step:
 13.15  5.27  1.31  9.45 -3.46 -4.21 -2.17  0.27  0.75 -0.89 -0.80 -0.43
  0.00  5.08  0.36  2.58 -0.96 -1.15 -4.21 -3.53 -3.40 -0.24 -0.20 -0.12
  0.00  0.00  3.85  2.12 -0.28 -0.42 -1.13 -0.68 -0.59  3.44  0.22  3.53
  0.00  0.00  0.00 -4.04 -1.31  0.92  1.28  0.77  0.67  0.46  0.18  0.37
  0.00  0.00  0.00  0.00  0.01  3.62 -3.62 -3.62 -3.62 -0.00 -3.62 -0.00
  0.00  0.00  0.00  0.00 -4.79  0.81 -2.46 -2.93 -3.02  0.42  0.16  0.33
  0.00  0.00  0.00  0.00  1.41 -1.61  0.67  0.40  0.35  0.24 -1.98  0.19
  0.00  0.00  0.00  0.00 -0.25  0.02 -0.12 -0.07 -0.06 -0.04  0.08 -0.03
  0.00  0.00  0.00  0.00  2.15 -0.74 -0.13 -0.84  0.05  3.21  0.32  0.32
```

Figure 4.31 : Output transcript of program execution (page 2)

```
Swapping rows 4 and 5
Swapping columns 4 and 4
Matrix after elimination step:
 13.15  5.27  1.31  9.45 -3.46 -4.21 -2.17  0.27  0.75 -0.89 -0.80 -0.43
  0.00  5.08  0.36  2.58 -0.96 -1.15 -4.21 -3.53 -3.40 -0.24 -0.20 -0.12
  0.00  0.00  3.85  2.12 -0.28 -0.42 -1.13 -0.68 -0.59  3.44  0.22  3.53
  0.00  0.00  0.00 -4.04 -1.31  0.92  1.28  0.77  0.67  0.46  0.18  0.37
  0.00  0.00  0.00  0.00 -4.79  0.81 -2.46 -2.93 -3.02  0.42  0.16  0.33
  0.00  0.00  0.00  0.00  0.00  3.62 -3.62 -3.62 -3.62  0.00 -3.62  0.00
  0.00  0.00  0.00  0.00  0.00 -1.37 -0.06 -0.46 -0.54  0.36 -1.93  0.29
  0.00  0.00  0.00  0.00  0.00 -0.03  0.01  0.08  0.10 -0.06  0.07 -0.05
  0.00  0.00  0.00  0.00  0.00 -0.37 -1.23 -2.15 -1.31  3.40  0.40  0.47

Swapping rows 5 and 5
Swapping columns 5 and 10
Matrix after elimination step:
 13.15  5.27  1.31  9.45 -3.46 -0.80 -2.17  0.27  0.75 -0.89 -4.21 -0.43
  0.00  5.08  0.36  2.58 -0.96 -0.20 -4.21 -3.53 -3.40 -0.24 -1.15 -0.12
  0.00  0.00  3.85  2.12 -0.28  0.22 -1.13 -0.68 -0.59  3.44 -0.42  3.53
  0.00  0.00  0.00 -4.04 -1.31  0.18  1.28  0.77  0.67  0.46  0.92  0.37
  0.00  0.00  0.00  0.00 -4.79  0.16 -2.46 -2.93 -3.02  0.42  0.81  0.33
  0.00  0.00  0.00  0.00  0.00 -3.62 -3.62 -3.62 -3.62  0.00  3.62  0.00
  0.00  0.00  0.00  0.00  0.00  0.00  1.87  1.47  1.39  0.36 -3.30  0.29
  0.00  0.00  0.00  0.00  0.00  0.00 -0.06  0.01  0.02 -0.06  0.05 -0.05
  0.00  0.00  0.00  0.00  0.00  0.00 -1.63 -2.55 -1.70  3.40  0.02  0.47

Swapping rows 6 and 8
Swapping columns 6 and 9
Matrix after elimination step:
 13.15  5.27  1.31  9.45 -3.46 -0.80 -0.89  0.27  0.75 -2.17 -4.21 -0.43
  0.00  5.08  0.36  2.58 -0.96 -0.20 -0.24 -3.53 -3.40 -4.21 -1.15 -0.12
  0.00  0.00  3.85  2.12 -0.28  0.22  3.44 -0.68 -0.59 -1.13 -0.42  3.53
  0.00  0.00  0.00 -4.04 -1.31  0.18  0.46  0.77  0.67  1.28  0.92  0.37
  0.00  0.00  0.00  0.00 -4.79  0.16  0.42 -2.93 -3.02 -2.46  0.81  0.33
  0.00  0.00  0.00  0.00  0.00 -3.62  0.00 -3.62 -3.62 -3.62  3.62  0.00
  0.00  0.00  0.00  0.00  0.00  0.00  3.40 -2.55 -1.70 -1.63  0.02  0.47
  0.00  0.00  0.00  0.00  0.00  0.00  0.00 -0.04 -0.01 -0.10  0.05 -0.04
  0.00  0.00  0.00  0.00  0.00  0.00  0.00  1.74  1.57  2.05 -3.30  0.24
```

Figure 4.32 : Output transcript of program execution (page 3)

```
Swapping rows 7 and 8
Swapping columns 7 and 10
Matrix after elimination step:
 13.15  5.27  1.31  9.45 -3.46 -0.80 -0.89 -4.21  0.75 -2.17  0.27 -0.43
  0.00  5.08  0.36  2.58 -0.96 -0.20 -0.24 -1.15 -3.40 -4.21 -3.53 -0.12
  0.00  0.00  3.85  2.12 -0.28  0.22  3.44 -0.42 -0.59 -1.13 -0.68  3.53
  0.00  0.00  0.00 -4.04 -1.31  0.18  0.46  0.92  0.67  1.28  0.77  0.37
  0.00  0.00  0.00  0.00 -4.79  0.16  0.42  0.81 -3.02 -2.46 -2.93  0.33
  0.00  0.00  0.00  0.00  0.00 -3.62  0.00  3.62 -3.62 -3.62 -3.62  0.00
  0.00  0.00  0.00  0.00  0.00  0.00  3.40  0.02 -1.70 -1.63 -2.55  0.47
  0.00  0.00  0.00  0.00  0.00  0.00  0.00 -3.30  1.57  2.05  1.74  0.24
  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.01 -0.07 -0.02 -0.04

Swapping rows 8 and 8
Swapping columns 8 and 9
Matrix after elimination step:
 13.15  5.27  1.31  9.45 -3.46 -0.80 -0.89 -4.21 -2.17  0.75  0.27 -0.43
  0.00  5.08  0.36  2.58 -0.96 -0.20 -0.24 -1.15 -4.21 -3.40 -3.53 -0.12
  0.00  0.00  3.85  2.12 -0.28  0.22  3.44 -0.42 -1.13 -0.59 -0.68  3.53
  0.00  0.00  0.00 -4.04 -1.31  0.18  0.46  0.92  1.28  0.67  0.77  0.37
  0.00  0.00  0.00  0.00 -4.79  0.16  0.42  0.81 -2.46 -3.02 -2.93  0.33
  0.00  0.00  0.00  0.00  0.00 -3.62  0.00  3.62 -3.62 -3.62 -3.62  0.00
  0.00  0.00  0.00  0.00  0.00  0.00  3.40  0.02 -1.63 -1.70 -2.55  0.47
  0.00  0.00  0.00  0.00  0.00  0.00  0.00 -3.30  2.05  1.57  1.74  0.24
  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00 -0.07  0.01 -0.02 -0.04

Log_10(1/error) in final matrix:
  7.84  8.03  8.79  7.52  7.07  8.06  8.56  8.13  8.16  8.77  8.57  8.95
   inf  8.32  8.94  7.94  7.61  8.57  8.88  8.46  8.51  8.92  8.89  9.15
   inf   inf  8.87  8.02  7.76  8.71  8.86  8.51  8.32  8.52  8.50  9.04
   inf   inf   inf  7.92  7.70  8.63  8.44  8.43  8.21  8.41  8.39  8.50
   inf   inf   inf   inf  7.65  8.58  8.30  8.22  8.01  8.23  8.20  8.36
   inf   inf   inf   inf   inf  9.28  9.23  9.17  9.02  9.06  9.05  9.24
   inf   inf   inf   inf   inf   inf  7.66  7.54  7.26  7.30  7.29  7.83
   inf   inf   inf   inf   inf   inf   inf  7.54  7.22  7.24  7.21  7.84
   inf   inf   inf   inf   inf   inf   inf   inf  7.74  7.78  7.74  8.43

Matrix verified full rank
```

Figure 4.33 : Output transcript of program execution (page 4)

Bibliography

[Boost] Boost C++ Libraries. http://www.boost.org

[Earle] C.Earle. H.E. Rauch - Function Theorist *Differential Geometry and Complex Analysis pp 15-31* Spinger-Verlag (1985)

[Hoffman09] D.Hoffman, M.Weber, M.Wolf. An embedded genus-one helicoid *Annals of Mathematics 169-2, pp 347-448* (2009)

[Kapouleas90] N. Kapouleas. Complete constant mean curvature surfaces in Euclidean three-space. *Annals of Mathematics 131, pp 239-330*(1990)

[Kapouleas96] N. Kapouleas. On designularizing the intersections of minimal surfaces. *Proceedings of the 4th International Congress of Geometry (Thessaloniki, 1996), pp 34-41* (1996)

[Karcher88] H. Karcher. Embedded minimal surfaces derived from Scherk's example. *manuscripta mathematica 62 pp 83-114* (1988)

[Manteuffel81] T. Manteuffel. An interval analysis approach to rank determination in linear least squares problems. *SIAM Journal of Science and Statistical Computing 2-3, pp 335-348* (1981)

[Meeks93] W.H.Meeks, H. Rosenberg. The geometry of periodic minimal surfaces. *Commentarii Mathematici Helvetici 68, pp 538-578* (1993)

[Moore79] R.E.Moore, F. Bierbaum. Methods and applications of interval analysis. *SIAM* (1979)

[Neumaier90] A. Neumaier. Interval methods for systems of equations. *Cambridge University Press* (1990)

[Osserman86] R. Osserman. Global properties of minimal surfaces in E^3 and E^n. *Annals of Mathematics 80, pp340-364* (1964)

[Traizet96] M. Traizet. Construction de surfaces minimales en recollant des surfaces de Scherk. *Annales de l'Institut Fourier 46, pp 1385-1442*(1996)

[Traizet00] M. Traizet. Weierstrass representation of some simply-periodic minimal surfaces. *Annals of Global Analysis and Geometry 20, pp 77-101*(2001)

[Weber98] M.Wever, M.Wolf. Minimal surfaces of least total curvature and moduli spaces of plane polygonal arcs. *Geometry and Functional Analysis 8, pp 1129-1170* (1998)

[Weber02] M.Weber, M.Wolf. Teichmuller theory and handle addition for minimal surfaces. *Annals of Mathematics 156, pp 713-795* (2002)